Design of Massive Optical Interconnection System for Data Centre Network

MUHAMMAD IMRAN

STUDENT NUMBER: 12210516

A Dissertation submitted in fulfilment of the requirements for the award of Doctor of Philosophy (Ph.D.)

SCHOOL OF ELECTRONIC ENGINEERING

DUBLIN CITY UNIVERSITY

Supervisor: Dr. Martin Collier

Co. Supervisor: Dr. Pascal Landais

May, 2017

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: 

Candidate ID No: 12210516

Date: 25/05/2017

Acknowledgement

In the Name of Allah, the Most Beneficent, the Most Merciful. First of all, I thank Almighty Allah, for giving me the strength to carry on this project and for blessing me with many great people who have been my greatest support in both my personal and professional life.

I would like to express my sincere gratitude to my supervisor Dr. Martin Collier and my co supervisor Dr. Pascal Landais for their continuous support in technical and non-technical matters related to my Ph.D studies, research work and thesis writing. I am also very thankful to Dr. Kostas Katrinis for his guidance.

I thank my all fellow lab mates for their help and support. Due to their company, my stay at Dublin City University was comfortable and enjoyable.

Last but not the least, I would like to pay my humble but full of emotion gratitude to my parents and Shaikh without their prayers and assistance this would not have been possible. I would also like to thank my entire family for providing me courage which I required most of the time during this work. I am very thankful to my loving wife, my lovely daughters Fatima and Momina, and my sons Hassan and Hussain for their love, support and patience.

CONTENTS

LIST OF ABBREVIATIONS

AWG Arrayed Waveguide Grating

AWGR Arrayed Waveguide Grating Router

CRC Cyclic Redundancy Check

CAGR Compound Annual Growth Rate

DCN Data Centre Network

FDL Fibre Delay Line

FPGA Field Programmable Gate Array

FTO False Timeout

FWC Fixed Wavelength Converter

HOL Head Of Line

HOSA Hybrid Optical Switch Architecture

ICT Information and Communication Technologies

MEMS Micro-Electro-Mechanical System

MZI Mach-Zehnder Interferometer

NIC Network Interface Card

O-E-O Optical-Electrical-Optical

OBS Optical Burst Switching

OCS Optical Circuit Switching

OFDM Orthogonal Frequency Division Multiplexing

OPS Optical Packet Switching

OSNR Optical Signal to Noise Ratio

OXC Optical Cross Connect

ROADM Reconfigurable Optical Add/Drop Multiplexer

RTT Round Trip Time

RTO Retransmission Timeout

SFP+ Small Form Factor Pluggable Plus

SOA Semiconductor Optical Amplifier

TCP Transmission Control Protocol

TDM Time Division Multiplexing

TWC Tunable Wavelength Converter

TDS Traffic Demand Scheduling

ToR Top of the Rack

VOQ Virtual Output Queue

WC Wavelength Converter

WDM Wavelength Division Multiplexing

WSS Wavelength Selective Switch

WXC Wavelength Cross Connect

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

Design of Massive Optical Interconnection System for Data Centre Network

Muhammad Imran

Abstract

Effective optical interconnect is a fundamental requisite to realize Internet-scale data centres due to the capabilities and benefits of optical devices. Optical interconnects are energy efficient and support massive bandwidths. The performance of optical interconnects is directly related to the type of optical switches and optical switching techniques used. Optical switches may be categorized into slow optical switches and fast optical switches. The optical switching techniques are Optical Circuit Switching (OCS), Optical Packet Switching (OPS) and Optical Burst Switching (OBS).

This thesis presents three novel optical interconnection schemes which are based on optical burst switching. These schemes are called Hybrid Optical Switch Architecture (HOSA), HOSA with Traffic Demand Scheduling (TDS) and Fast Optical Switch Architecture (FOSA). The first two schemes are based on a hybrid design that utilizes fast and slow optical switches while the third scheme is based on using only fast optical switches. The proposed schemes consider OBS with a two-way reservation protocol that ensures zero burst loss. In the two-way reservation protocol, the connection is established for each burst before transmission. To evaluate the performance of these schemes, network-level simulation is used.

The proposed architectures consider a single stage core topology that can be easily scaled up (in capacity) and scaled out (in the number of racks) without requiring major re-cabling and network reconfiguration. The proposed schemes feature separate data and control planes. The control plane comprises a centralized controller while the data plane contains an array of optical switches. A scalability analysis of the proposed topology is presented and shows that this topology is scalable to hundreds of thousands of servers. This thesis also presents a trade-off between cost and power consumption of the proposed designs by comparing them with conventional interconnects using analytical modelling.

In HOSA, the key idea is to route high volume traffic through a fast optical switch during the reconfiguration of a slow optical switch. The traffic is moved to the slow optical switch once it is reconfigured. The aggregated traffic should be large enough so that it can bypass the slow optical switch during its reconfiguration phase. This technique hides the reconfiguration time of slow optical switch but the latency introduced due to burst aggregation is still high. In HOSA with TDS, small bursts are considered to reduce the latency of burst aggregation. The controller in HOSA with TDS technique maintains a traffic demand matrix which updates traffic demand on

periodic intervals and assigns slow paths for the high traffic volume. A resource allocation algorithm is proposed that allocates paths of fast and slow optical switches efficiently.

In HOSA with TDS, the control plane can only support applications that have high traffic stability. So for dynamically changing communication patterns, this thesis presents a new design called FOSA. The FOSA is based on only fast optical switches and it also uses OBS with the two-way reservation i.e. there is no additional burden on the control plane for maintaining traffic demands as was done in HOSA with TDS. The proposed technique shows considerable improvement in terms of throughput and packet loss ratio as compared to the traditional methods of OBS while performance comparable in terms of delay with the traditional methods of OBS is also achieved. The proposed technique also demonstrates performance comparable to that of electrical data centre networks.

The performance of TCP over traditional OBS network is degraded by the bursts losses due to contention even at a low traffic load. In FOSA, the performance of the TCP is evaluated. Since in the proposed scheme, the burst loss is zero due to two way reservation, the results show significant improvement of TCP performance in terms of throughput, time and packets loss as compared to the traditional methods of OBS and the conventional electronic packet switching data centre networks for all types of workloads.

LIST OF PUBLICATIONS

This work is based on the following contributions either published or in review phase at various places.

Journal Publications

1. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Software-Defined Optical Burst Switching for HPC and Cloud Computing Datacenters", *Journal of Optical Communications and Networking*, 8.8 (2016): 610-620. **(Impact Factor: 2.18)**.

2. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Performance Evaluation of TCP over Software-Defined Optical Burst-Switched Data Center Network", *invited paper submitted to Journal of Computational Science*, **(Impact Factor: 1.078)**.

3. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Performance Evaluation of Hybrid Optical Switch Architecture for Data Center Networks"", *Journal of Optical Switching and Networking*, **Best paper award**, 21 (2016): 1-15, **(Impact Factor: 1.137)**.

4. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Software-Controlled Next Generation Optical Circuit Switching for HPC and Cloud Computing Datacenters", *Electronics*, 4.4 (2015): 909-921.

Conference Publications

1. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Performance Evaluation of TCP over Optical Burst-Switched Data Center Network", *In Computational Science and Engineering (CSE), 2015 IEEE 18th International Conference on*, pp. 51-57. IEEE, 2015.

2. **Muhammad Imran**, Pascal Landais, Martin Collier,and Kostas Katrinis, "A data center network featuring low latency and energy efficiency based on all optical core interconnect", *In Proceedings of the 17th IEEE International Conference on Transparent Optical Network (ICTON)*, vol., no., pp.1,4, 5-9 July 2015, doi: 10.1109/ICTON.2015.7193537.

3. **Muhammad Imran**, Pascal Landais, Martin Collier,and Kostas Katrinis, "Performance Analysis of Optical Burst Switching with Fast Optical Switches for Data Center Networks", *In Proceedings of the 17th IEEE International Conference on Transparent Optical Network (ICTON)*, vol., no., pp.1,4, 5-9 July 2015, doi: 10.1109/ICTON.2015.7193596.

4. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "HOSA: Hybrid Optical Switch Architecture for Data Center Networks", *In Proceedings of the 12th ACM International Conference on Computing Frontiers (CF'15)*, p.27, doi: 10.1145/2742854.2742877.

5. **Muhammad Imran**, Martin Collier, Pascal Landais and Kostas Katrinis, "Energy Efficient Data Center Network based on Slow and Fast Optical Switches", *In Photonics Ireland Conference*,2015.

CHAPTER 1

MOTIVATION

Internet traffic has been increasing exponentially over the last few years due to the emergence of new end user applications which are based on cloud computing infrastructure. These applications run on the servers deployed in the data centres and require huge network bandwidths. The major infrastructure of today's Internet comprises data centres. Data centres provide a number of services from social networking to large-scale scientific calculations. They can be defined as big centres of storage and computing resources that communicate extensively with each other to serve the ever-increasing demands of customers [10].

The data centres are getting more and more importance in our lives because the cloud computing has shifted computation and storage away from desktops to large-scale data centres [11]. With the advancements of smart mobile devices and increasing bandwidth demands for multimedia applications and other data services, IP traffic will keep on increasing at an exponential pace [12, 13].

Apart from the demands of increasing bandwidth, it is predicted that the information and communication technologies (ICT) industry will contribute 2 to 3 percent of global greenhouse gas emissions, a share that is quickly increasing [14]. Data centres,

being ICT infrastructure need to be evaluated in terms of interconnection design to support very large scales while leading to ultimate footprint and power savings.

1.1 Traditional Data Centres Architecture

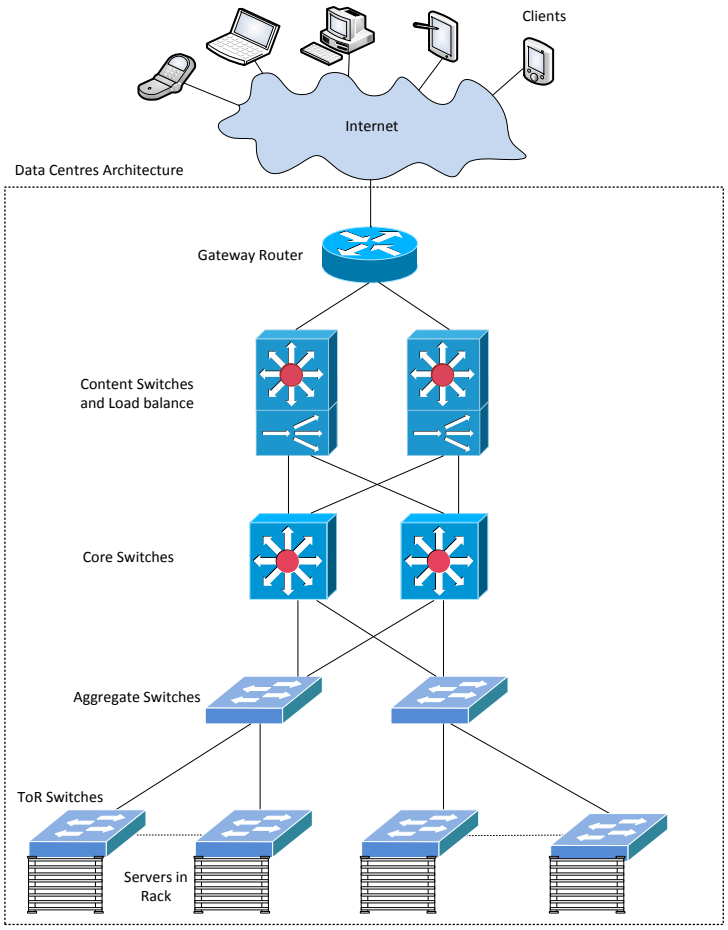


Figure 1.1. Data Centre Architecture.

There are various types of data centres such as computational, storage and high performance computing (HPC) etc. This thesis targets computational data centres. The traditional architecture of the computational data centre network (DCN) is based on a hierarchical design as shown in Figure 1.1. It features several layers of electrical switches. At the front end, the content and load balance switches are connected to the internet through the gateway routers, while at the back end, they are linked to the core switches. The core switches are linked to the aggregate switches and the aggregate

switches are connected to the Top of the Rack (ToR) switches. Each ToR switch is connected to the servers in the rack. All the switches feature an electronic switch fabric and the links between them can be either copper cables or optical fibres. In the case of optical fibre links, optical-electrical-optical (O-E-O) conversion is required at every port of the switch. When a request comes from the external network, it first comes to the load balance and content switches which route the request to the appropriate servers. To fulfil the request, the servers can coordinate with other servers within the same or different racks. For example, the application servers can coordinate with the database servers to process the request. After completing the request, the response is sent to the external network through the gateway routers.

Table 1.1. Performance , Power and Cost Requirements for Data Centres [9], [2]

Year	Peak Performance	Power Consumption	Equipment Cost	Bandwidth
2012	10PF	5MW	$225M	1Pbytes/s
2016	100PF	10MW	$350M	20Pbytes/s
2020	1000PF	20MW	$500M	400Pbytes/s

1.2 Limitations of Traditional DCNs

There are significant challenges to meet the growing performance requirements with current computational data centre architectures. These are described below.

1.2.1 Power

The electrical switches at different layers of a DCN and the transceivers required for O-E-O conversion are significant sources of power consumption in traditional DCN designs. The power consumption of the current interconnection network incurs 23% of the total IT power consumption in a DCN as shown in Figure 1.2 while it is predicted that the interconnection network will incur a much higher percentage of overall IT power consumption in future DCNs [15]. In traditional DCNs, the power consumption of one port of aggregate/core switch is higher than the power consumption of one port

of ToR switch. For example, the power consumption of one port of ToR switch is 3.7 watt [16] while it is 12.4 watt for aggregate/core switch [17]. Exact proportion of power consumption for ToR, aggregate and core switches varies with the number of ports of these switches in DCN. It is shown in Table 1.1 that the peak performance required of data centres will continue to rise tremendously but the affordable budget for the total permissible power dissipation by the data centres is increasing at a much slower rate i.e. it doubles every 4 years due to various thermal dissipation factors.

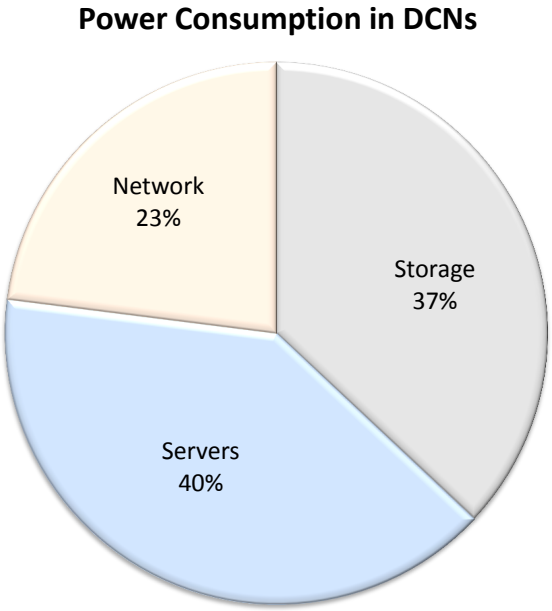


Figure 1.2. Power Consumption in DCNs [1]

The power dissipation of the data centres also has a significant effect on the environment. It has been reported in 2007 that data centres accounted for 14% of the total ICT greenhouse gases emissions and it is expected to grow up to 18% by 2020 [18].

1.2.2 Traffic Locality

The projection of traffic growth in data centres according to the Cisco cloud index [2] is shown in Figure 1.3. Observe that during the period from 2014 to 2019, the majority of data centre traffic will remain within the data centre while only a small portion of the traffic will go to the external network. Some of the traffic will also

be exchanged between data centres for distributed and replicated services between databases in different data centres. Due to this high traffic locality, high bandwidth and low latency interconnections are required.

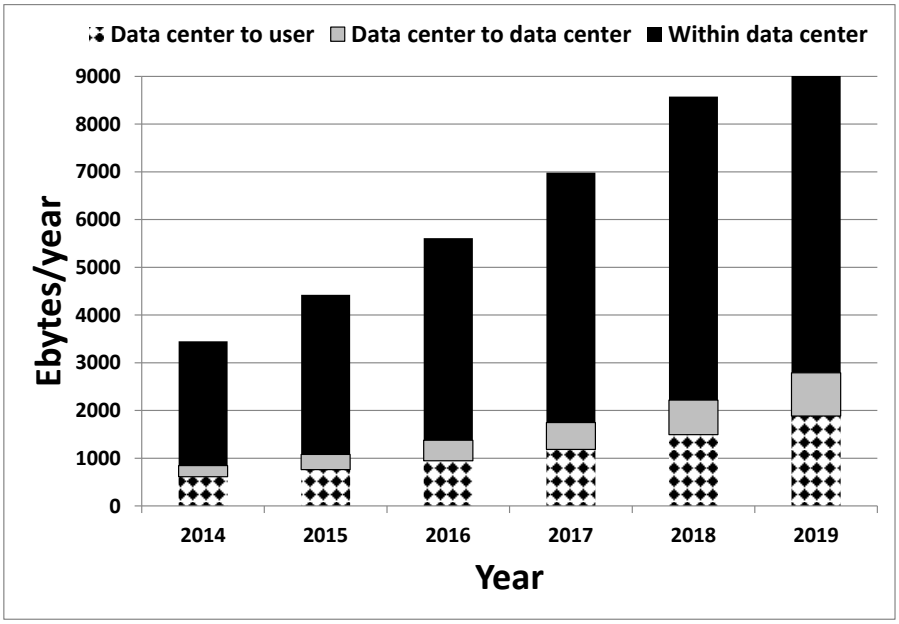


Figure 1.3. Traffic Movement in Data Centres from 2014 to 2019 [2]

1.2.3 Higher Bit Rates

The performance of communication systems at high data rates using electrical transmission lines is degraded by dielectric losses and losses incurred due to skin effect. Power dissipation increases as data rates increase in electrical transmission lines. For example, for 10 Gig E, power restrictions limit cable length to about 10 m. Longer cables are possible, but power can exceed 6 W/port which is not feasible in large-scale data centres [19]. On the other hand, there is no power consumption in optical fibre because it is a passive device. However, there is attenuation in optical fibre but it is independent of the data rate. Higher data rates in optical fibre can be achieved by engineering a better laser or by using digital signal processing. For example, University College London (UCL) researchers have achieved 1.125 terabits per second over fibre optic by using advanced digital signal processing algorithms [20].

1.2.4 Scalability

An architecture based on ethernet links and switches (with limited capacity) will be very challenging to manage as the traffic flowing within data centres continues to grow exponentially. The hierarchical design cannot support the growing traffic in data centres that will be equipped with more and more servers and also more microprocessor cores per server. The future data centres must have the capacity to incorporate hundreds of thousands of servers. Large cloud computing data centres owned by Amazon, Microsoft and Google have tens of thousands of servers [21]. With the expected growth in data center traffic, the number of servers in data centres is destined to increase which poses a significant challenge to the data center interconnection network.

1.2.5 Latency

Latency is introduced by queuing in buffers and by propagation delays incurred by packets during transmission from one node to another. Packets have to be buffered by switches during packet processing and this delay can be large when there is congestion in the network. Although the switching speed of electronic switches is of the order of micro- or nano-seconds, overall end-to-end packet delay is significant and will need to be reduced in future data centres.

1.2.6 Oversubscribed Network

Traditional data centres networks are oversubscribed. For example, if 40 servers in a rack are connected with 1 Gbps link to the ToR switch and the ToR switch is connected by a 10 Gbps link to the aggregate switch, the network is oversubscribed by a 4:1 ratio. In the worst case scenario, if all the servers within the same rack generate 1 Gbps data to communicate with the servers in other racks, the 10 Gbps link will not have enough capacity to forward the 40 Gbps of data. There are various approaches proposed to solve the issue of over-subscription. One approach is to use intelligent workload placement algorithms to allocate network-bound service components to physical hosts with high bandwidth connectivity [22]. An alternative approach is to flexibly allocate

more network bandwidth to service components with heavy communications. Thus, even for a few thousand servers, uniformly high capacity networks appear to be an overkill. As the size of the network grows, this weighs on the cost, power consumption and complexity of such networks.

1.2.7 Performance

The performance of data centres has been increasing on the order of 10 times every 4 years with bandwidth increasing on the order of 20 times in the same interval as shown in Table 1.1. Power consumption can only be allowed to increase perhaps twofold and cost by a factor of 1.5.

1.3 Benefits of Optical Interconnects

An optical interconnection system can meet the above mentioned challenges due to the properties of optical components. Optical switches are power efficient and consume less power than electrical switches. For example, electronic packet switches consume 12.5 W power per port which is 50 times higher than micro-electro-mechanical system (MEMS) based optical switches which only consume 0.24 W power per port while the cost per port of both type of switches is approximately the same, i.e. $500 [23].

Optical interconnects can provide high capacity links to meet requirements for traffic locality and higher bit rates by using optical fibres and optical transceivers. Researchers from Bell Labs successfully sent data at the speed of 31 Terabits-per-second over 7200km in an experiment done at Alcatel-Lucent's Innovation City campus in Villarceaux near Paris [24]. There is an increasing trend for using 10 Gig Ethernet network interface cards (NICs) in servers. Google has 10 Gig E deployments and is pushing the market for 40/100 Gig E [25] and this is only feasible with optical fibres. In the near future, higher bandwidth transceivers are going to be adopted (for 40 Gbps and 100 Gbps Ethernet) such as 4×10 Gbps Quad Small Form Factor Pluggable(QSFP) modules with four 10 Gbps parallel optical channels and CXP modules with 12 parallel

10 Gbps channels. Higher data rates could also solve the oversubscription problem by providing full bisection bandwidth in fully subscribed network.

1.4 Challenges of Optical Interconnects

There are significant challenges to be addressed before designing an optical interconnection system for DCN. The performance of the optical interconnects is directly related to the type of optical switches and their underlying optical switching technologies. These are described below.

1.4.1 Optical Switches

Traditional optical switches are based on MEMS technology. The attractive features of MEMS switches include: a) excellent power efficiency due to the use of passive switching, b) high port density, c) low insertion loss and crosstalk, d) an absence of transceivers due to using all-optical switching, e) lower cost, f) support of bidirectional communication, and g) data rate independence. They are also highly scalable and are commercially available e.g. 3D-MEMS [26]. However, they have long switching times, of the order of tens of milliseconds. Fast optical switches using technologies such as arrayed waveguide grating routers (AWGRs) and semiconductor optical amplifiers (SOAs) are now available. An AWGR is a passive device and works in combination with tunable lasers (TLs) or tunable wavelength converters (TWCs). The switching time of these switches is determined by the tuning speed of TLs or TWCs which is of the order of a few nanoseconds [27]. An SOA works as an ON/OFF switch and also compensates for losses that occur during transmission of optical signals. SOA switches also have a switching time in the range of a few nanoseconds [28,29]. Although both types of switches are fast, they are expensive in comparison to MEMS switches of the same capacity. Wavelength Selective Switch (WSS) is an another type of optical switch that can distribute the incoming set of wavelengths to a different set of outgoing wavelengths. The switching time of WSSs is in the range of microseconds to

milliseconds [30, 31]. Further details about optical switches are provided in Chapter 2.

1.4.2 Optical Switching

Optical Switching techniques that exist in optical networks are optical circuit switching (OCS), optical packet switching (OPS) and optical burst switching (OBS). All these techniques have some advantages and disadvantages over one another.

The OCS is a connection oriented optical switching technique [32, 33]. In OCS, a connection is established before data transmission on a pre-defined path from the sender to the receiver. A control packet is sent in advance to establish a link to the destination. Once the link is established, an acknowledgement is sent back to the sender and then the data transmission starts. The OCS does not require transceivers since it does not employ conversion between light and electricity i.e. no O-E-O conversion is required at intermediate core switches as the link is already established. Large connection establishment time and bandwidth underutilization in the case of low traffic load are the major limitations of the OCS. The wavelength continuity constraint is another limitation of the OCS. However, wavelength converters are used to address this limitation. The OCS has been widely used in the long-haul backbone optical network for many years. In backbone optical network, the connections are established for a long duration. There is a very infrequent demand to change the connection setup. For example, the connections between cities and countries in the long-haul optical network last for days and weeks. This has made OCS the preferred choice for the backbone optical network. MEMS switches are ideal switches to use with OCS because their long switching time does not limit performance.

The OPS is a connectionless optical switching technique [34–37]. In OPS, a packet consists of a data and a header portion which are in the optical domain. When the packet arrives at the optical switch, the header is removed from the packet and is converted into the electrical domain for processing. During this processing time, the data in the packet has to be buffered in the node. Fibre delay lines (FDLs) are used for this purpose which can provide limited buffering by routing the light to the appropriate

fibres. The packet is dropped if the switch is not configured within this time. The processing speed of the switch controller should be compatible with the data rate of the incoming channel. Low processing speed of the switch controller also results in packet losses. The limitations of OPS are the lack of feasible optical buffers and packet losses in the case of output port contention or low processing speed of the switch controller. Fast optical switches are required to use OPS because of their low switching time.

The OBS [38–42] is different from OCS and OPS and is considered as a compromise between them. It has a separate control and data plane similar to OCS. Packets are aggregated into a burst. A control packet is then transmitted on a dedicated control channel to reserve resources on all intermediate nodes from the source to the destination. The burst is sent at a particular time after sending the control packet which is called the offset time. During the offset time, the burst is temporarily stored at the edge node before transmission. During this time, the switch controller at the core node processes the control information and sets up the switching matrix for the incoming burst. Burst loss due to output port contention is the major limitation of the OBS network. Output port contention can occur due to unavailability of a wavelength at the desired output port for an incoming burst. Several techniques exist in the literature to avoid contention such as FDLs [43, 44], Deflection Routing [45, 46], Wavelength Conversion [47] and Segmentation Based Dropping [48, 49] but none of them can guarantee zero burst loss. OBS with a two-way reservation scheme ensures zero burst loss in which a control packet reserves resources across all intermediate nodes from the source to the destination and is sent back to the source as an acknowledgement. However, the control packet has a high Round trip time (RTT) for a large wide area optical network. This is because of the high propagation and switching delay.

1.5 Overview of Proposed Solution

In this thesis, novel optical interconnection schemes which are based on OBS are proposed. Historically, the OBS was proposed for the backbone optical core network but

it has not replaced OCS due to its limitation of high burst loss in this application. The proposed schemes consider OBS with a two-way reservation protocol that ensures zero burst loss. In the two-way reservation protocol, the connection is established for each burst before transmission. The two-way reservation is not suitable for long-haul backbone optical networks due to the high RTT of the control packet but for the proposed optical interconnect for the DCN, this RTT is not high for several reasons:1) the propagation delay is negligible, 2) faster optical switches are used at the core, 3) a fast optical control plane is used, 4) processing of the control packet is rapid and 5) a single hop topology is used. Network-level simulation is used to evaluate the performance of the proposed schemes.

In the first scheme, an optical interconnection scheme which is based on both fast and slow optical switches is proposed which is called hybrid optical switch architecture (HOSA). The proposed technique leverages strengths of both types of optical switches. The strengths of one type of optical switch compensate for the weaknesses of the other type. HOSA employs a single stage core topology that can be easily scaled up (in capacity) and scaled out (in the number of racks) without requiring major re-cabling and network reconfiguration. HOSA features separate data and control planes. The control plane comprises a centralized controller while the data plane contains an array of fast and slow optical switches. The main idea is to route high volume traffic through the fast optical switch during the reconfiguration of the slow optical switch. The traffic is moved to the slow MEMS switch once it is reconfigured. This results in the overall switching time being that of the fast optical switch. A resource assignment algorithm is presented that allocates resources efficiently to ensure minimum latency. The scalability of the proposed design is evaluated by considering various capacities of servers in a rack and various ratios of fast and slow optical switches. A trade-off between cost and power consumption of the proposed design using analytical modelling to compare it with conventional interconnects is also presented. Furthermore, the trade-off between the performance and the capacity of both types of optical switches is evaluated. In this scheme, large burst aggregation time is the major limitation i.e. the aggregated traffic should be high enough so that it can bypass the MEMS switch during its reconfiguration time.

In the second scheme, HOSA supplemented with a Traffic Demand Scheduling scheme is proposed to overcome the limitation of high aggregation time in HOSA. In HOSA with TDS, there is no need to aggregate a large amount of traffic. The controller maintains a traffic demand matrix which updates traffic demand on periodic intervals and assigns slow paths for the high traffic volume. A resource allocation algorithm is proposed in the control plane for efficient utilization of the resources that results in high throughput and low latency. However, similar to other hybrid techniques [23, 50–54], HOSA with TDS has the limitation of the control plane overhead to measure the communication patterns and calculate a new schedule. As a result, the control plane is limited to support only traffic with high stability. Stability refers to the duration of the traffic flows i.e. high stability refers to the traffic flows that last several seconds. Due to the limitation of the control plane, HOSA with TDS does not perform well for dynamically changing communication patterns. To overcome this limitation, this thesis presents a new optical interconnect called fast optical switch architecture (FOSA).

FOSA is based on only fast optical switches. In this design, only OBS with the two-way reservation is used. In FOSA, there is no extra processing overhead on the control plane for maintaining traffic demands as was done in the second scheme. This scheme works well for dynamically changing communication patterns. The proposed technique shows considerable improvement in terms of throughput and packet loss ratio compared to the traditional methods of OBS while delays comparable to those of the traditional methods of OBS are also achieved. The proposed technique also demonstrates performance comparable to that of electrical data centre networks.

Another reason that OBS has not replaced OCS in the long-haul optical network is because of its poor performance with TCP. The performance of TCP over a traditional OBS network is degraded by the wrong interpretation of congestion in the network. The burst losses due to contention can be misinterpreted by the burst losses due to congestion. The contention refers to the burst loss due to unavailability of a wavelength even at the low traffic load in the network. The performance of the TCP is evaluated on FOSA. As in the proposed scheme, the burst loss is zero due to the two-way reservation protocol; the results show significant improvement of TCP performance in

terms of throughput, time and packets loss as compared to the traditional methods of OBS. The proposed scheme also demonstrates more efficient TCP performance than the conventional electronic packet switching DCN for various types of workloads.

1.6 Thesis Structure

This section provides an overview of the later chapters of this thesis. The next chapter is a study of the existing efforts of the research community to resolve the challenges posed by data centre networks. The technical contribution of this research is documented in the next four chapters and the last chapter contains conclusions.

Chapter 2 starts with the brief overview of various types of optical devices that could be of interest in designing an optical interconnection system for DCNs. Afterwards, the detailed description of various types of optical switches used in optical networks is presented. Various optical interconnects schemes that have been proposed for data centre networks are discussed and a brief overview of each architecture is provided. Finally, a comparative analysis of these optical interconnects is provided.

In chapter 3, the proposed design of the HOSA is presented. The detailed description of the component design and various algorithms which are used in the ToR switch and in the control plane are discussed. A scalability analysis of the proposed interconnect, investigating various ratios of slow and fast optical switches, is provided. This chapter also presents an analysis of cost and power consumption of the proposed design, comparing it with other well-known interconnects. Furthermore, the performance of the system is evaluated using network-level simulation by considering various traffic workloads, and by using different capacities of slow and fast optical switches.

In chapter 4, HOSA with the TDS technique is discussed. The detailed description of the proposed resource allocation algorithm in the control plane is presented. The performance evaluation of the system using network-level simulation, considering diverse workload communication patterns and system design parameters, is also discussed.

Chapter 5 describes a new architecture FOSA which is based on using only fast optical switches. The detailed description of various algorithms used in the ToR and in the controller is provided. The scalability of the new design using only fast optical switches is discussed. The performance analysis is done by using network-level simulation, comparing the proposed technique with traditional OBS and electrical data centre network.

In chapter 6, the performance issues of TCP over OBS networks are discussed. The performance of TCP over FOSA using network-level simulation is presented and a comparative analysis with traditional OBS and electrical data centre network is done.

Finally, in chapter 7, an overview of the contributions, HOSA which is based on hybrid optical switch design, HOSA with TDS technique and FOSA design which is based on fast optical switches is given. Some future directions in the area of optical data centre are also discussed.

1.7 Summary of Contributions

Below is a list of technical contributions of this thesis for more clarity.

- Introducing a concept of two-way reservation scheme of OBS in data centre network.
- Proposing a scalable hybrid optical switch architecture HOSA, which is based on fast and slow optical switches using a single stage core technology.
- Performing scalability analysis of the HOSA.
- Performing cost and power consumption analysis of the HOSA and comparative analysis with other well know interconnects.
- Analytical modelling of cost and power consumption analysis of the HOSA and other well known interconnects.
- Investigating the performance of HOSA using network-level simulation.

- Proposing HOSA with TDS technique to improve HOSA.
- Evaluating the performance of HOSA with TDS using network-level simulation and performing a comparative analysis with baseline electrical data centre network.
- Introducing a new design FOSA which is based on only fast optical switches.
- Assessing the performance of the FOSA using network-level simulation and performing a comparative analysis with traditional OBS and with baseline electrical data centre network.
- Performing scalability analysis of the FOSA design using only fast optical switches.
- Evaluating the performance of TCP on the FOSA using network-level simulation and performing a comparative analysis with traditional OBS and with a baseline electrical data centre network.

CHAPTER 2

SURVEY OF OPTICAL INTERCONNECTS FOR DCN

2.1 Introduction

Data centres are experiencing an exponential increase in the amount of network traffic that they have to sustain due to the emergence of new end user applications which are based on cloud computing infrastructure. To meet the requirement of increasing bandwidth demand, large data centres are needed with thousands of servers interconnected with high bandwidth switches. The traditional DCNs, based on electronic packet switches, consume excessive power and are unable to meet the increasing bandwidth demand of emerging applications. Optical interconnects have gained significant attention in recent years because they can be used to address the challenges of traditional DCN. They can offer high throughput, low latency and low power consumption compared to the traditional DCNs.

This chapter presents a detailed survey on optical interconnects for DCNs which are categorized according to the type of optical switches used. Furthermore, it provides a

qualitative comparison of the proposed schemes based on their main features such as optical switching techniques, connectivity, scalability, cost and power consumption.

This chapter starts with the brief overview of various types of optical switches. Afterwards, various optical interconnects schemes that have been proposed for data centre networks are discussed and a general insight on each architecture is provided. In the end, a comparative analysis of these optical interconnects is given.

2.2 Optical Switches

Optical interconnects can be realized by using optical switches and other optical components. In this section, optical switches are categorized with relevant to their configuration/switching time. Optical switches are divided into two types: 1) Slow optical switches; and 2) Fast optical switches. Slow optical switches have a switching time in the millisecond scale while fast optical switches have switching time in the range of a few nanoseconds.

2.2.1 Slow Optical Switches

Slow optical switches are further divided into two types: 1) Micro electro mechanical system (MEMS) switch and 2) Wavelength selective switch.

MEMS Switch

The MEMS switch is the most common type of slow optical switch. It is also known as an optical circuit switch. It uses an $N \times N$ crossbar of mirrors to direct a beam of light from any input port to any output port where N is the number of input/output ports. A schematic of its design is shown in Figure 2.1. The mirrors are attached with small motors, each of which is about 1 mm^2. The built-in controller processor positions the mirrors in the bar or cross position to implement a desired connection matrix and receives remote requests to reconfigure the mirrors into a new connection

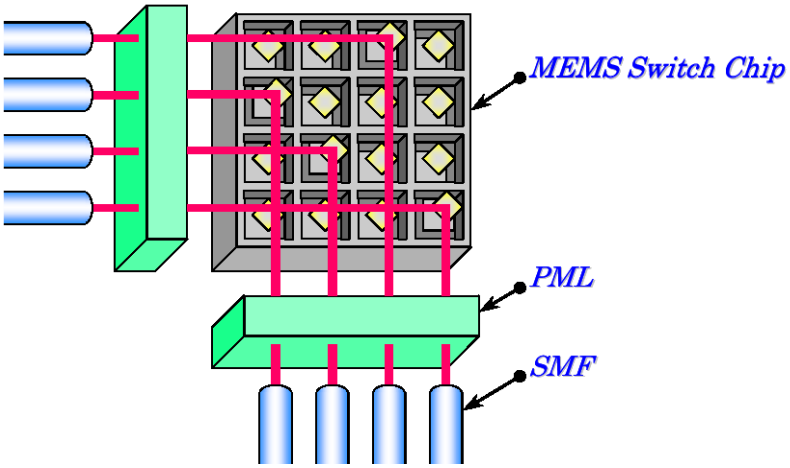


Figure 2.1. MEMS Switch [3]

matrix. Repositioning the mirrors to a particular position takes some time which is called reconfiguration/switching time [3, 55].

The attractive features of MEMS include: a) high port density, b) transceivers are not required due to all-optical switching, c) power efficiency due to passive switching, d) low insertion losses and crosstalk, e) data rate independent i.e. high link bandwidth can be achieved due to WDM technologies, f) support bidirectional communication, g) less expensive, and are based on mature technologies. All of these advantages come at the cost of switching time which is of the order of tens of milliseconds. These switches are mostly used in long-haul backbone optical network by using optical circuit switching for long-lived connections. MEMS switches with 384 ports (Polatis) and 320 ports (Calient) are commercially available while with as many as 1,000 ports being feasible and hope to be available in the future [26, 56]. Research prototype of MEMS optical switches has already scaled to more than a thousand input and output ports [57].

Wavelength Selective Switch

Wavelength Selective Switch (WSS) is a device that can divide the incoming set of wavelengths to a different set of outgoing wavelengths. It is shown in Figure 2.2. Each set of wavelength is destined to a specific output port. The reconfiguration time

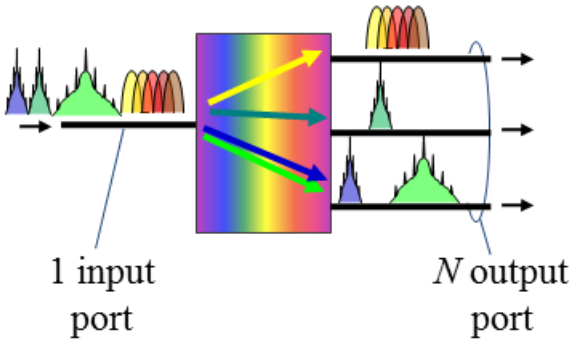


Figure 2.2. WSS Switch [4]

of WSSs has been reported in the range of a few milliseconds [30] while an other study has revealed it in the range of a few microseconds [31].

2.2.2 Fast Optical Switches

Fast optical switches are divided into three types: 1) AWGRs, 2) Photonic Space Switches and 3) SOA-Based Switches. Most of the optical interconnects using optical packet switching proposed in literature are based on AWGR and SOA based switches while photonic space switches combined with other optical switches also have the capacity to use in future DCNs. All these switches have the switching time in the range of a few nanoseconds. These switches are described below:

AWGR Switches

AWGs can also be used as a static WDM router called AWGR. AWGR is a combination N $(1 \times N)$ AWGs on the sender side and N $(N \times 1)$ AWGs on the receiving side arranged in a cyclic way where N is the number of input/output ports. It can provide strictly non-blocking switching, if it is used with TWCs at the sender side and FWCs at the receiver side [58]. Their design is shown in Figure 2.3. A TWC selects wavelength according to the desired output port. A TWC provides a switching time in the range of a few nanoseconds. An AWGR switch design with (512×512) ports has been reported by using 512 channels with 10 GHz channel spacing [59] while another study

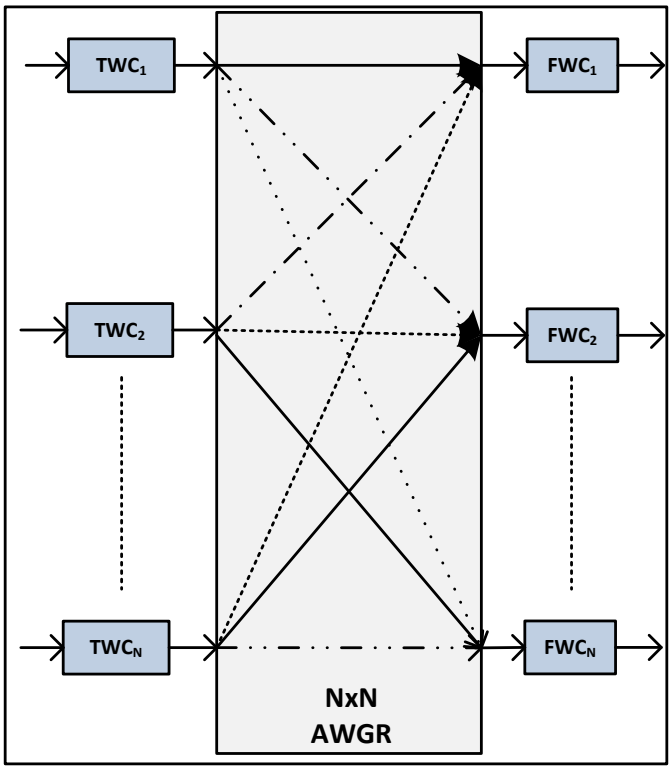


Figure 2.3. AWGR Switches

recently presented design, fabrication and characterization of (512×512) AWGR with a channel spacing of 25 GHz [60].

AWGR as a router module is commercially available in 32×32 standard configurations [61,62]. However, custom design with a large number of ports can also be made. For example, AWGs with 128 channels are commercially available [63], so a router module of AWGR with 128×128 configurations can be made by arranging them in a cyclic way as described above.

Photonic Space Switches

Photonic space switches with $(1 \times N)$ configurations have been built by using Polarized Lead Zirconium Titanate (PLZT) waveguide technology [64] where N is the number of input/output ports. These switches are made by using (1×2) Mach-Zehnder with 3dB couplers arranged in multiple stages to realize a $(1 \times N)$ switch architecture. These switches also provide a switching speed in the range of a few nanoseconds. Another

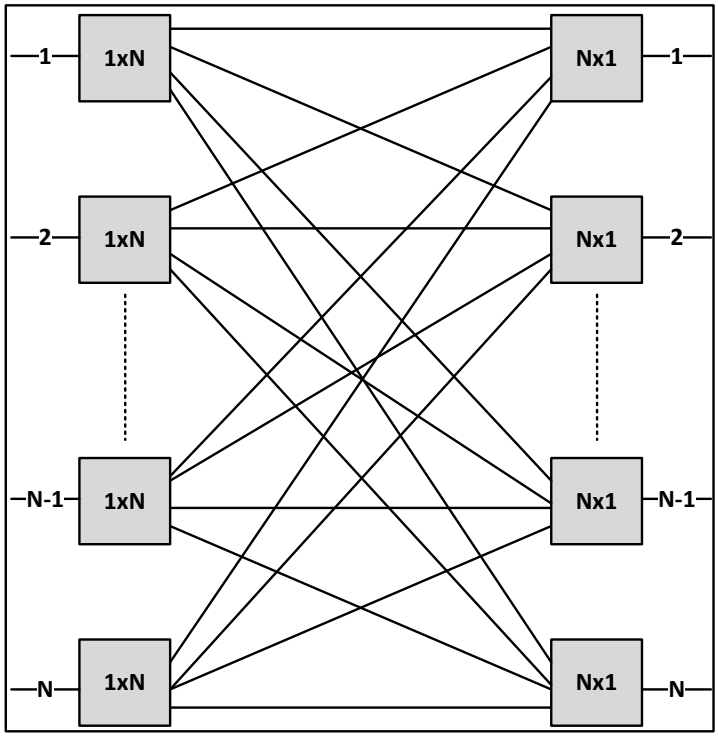


Figure 2.4. Photonic Space Switches

type of $(1 \times N)$ switches has been presented by using phased array switching technology [65]. These switches use a single stage of phased array modulators between star couplers instead of multiple stages of Mach-Zehnder modulators. An $(N \times N)$ strictly non-blocking switch fabric can be achieved by arranging these switches into two stages. An example of an $(N \times N)$ switch configuration is shown in Figure 2.4. The number of $(1 \times N)$ switches required for an $(N \times N)$ implementation is $2N$, half of them arranged in a $(1 \times N)$ configuration and the other half in an $(N \times 1)$ configuration. A very large switch fabric can be achieved by arranging these switches in multiple stages in conjunction with SOAs in order to make good the losses introduced in optical transmission system. Unlike AWGR, these switches can be used in bidirectional communication. Currently, switches with (1×16) configuration are commercially available [66].

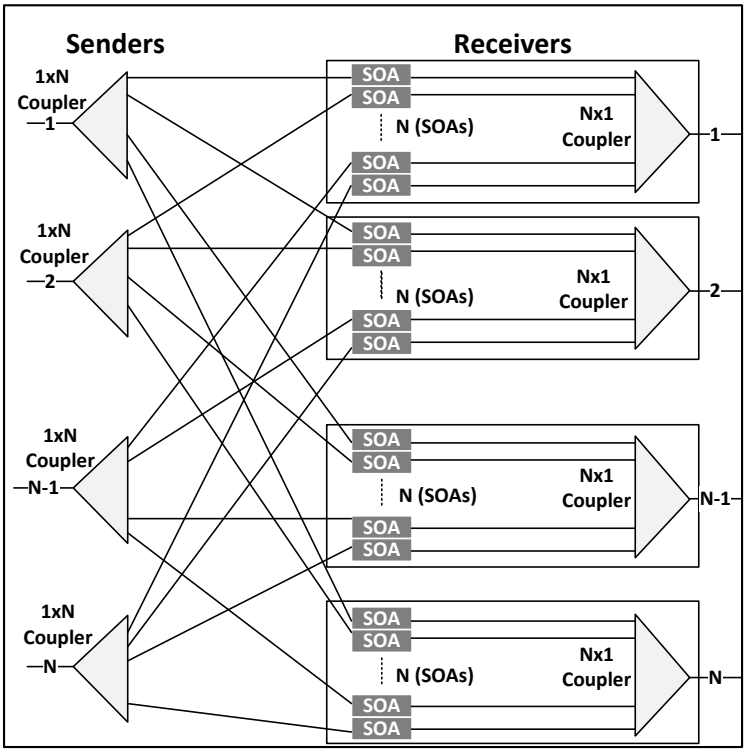


Figure 2.5. SOA-Based Switches

SOA-Based Switches

SOA-based switches are commonly realized using a broadcast-and-select architecture arranged in 3 stages as shown in Figure 2.5. First, the input signal is broadcast using a $(1 \times N)$ coupler. Each output of the $(1 \times N)$ coupler is attached to one of N SOAs per output port. The SOA is used as a gate element and also provides optical gain in order to make up for the losses introduced by insertion and coupling. There are two states of SOAs: ON state and OFF state. SOAs consume different levels of power in these two states. In the ON state, they provide gain to the optical signal while in the OFF state they work as a gate switch i.e. they absorb optical signal. It has been reported [29] that SOAs are required after every (1×32) coupling factor, in order to provide gain to overcome the losses. A large scale strictly non-blocking switch comprising (1024×1024) ports can be realized if two stages of SOA gates are used i.e. after the first stage of SOAs, light is broadcast again by using (1×32) coupler to another stage of SOAs. Optical Signal to Noise Ratio (OSNR) degradation is experienced by the optical signal after passing through more than 2 stages of SOAs. Proposed design of SOA-

based switch in [29] is based on a simulation work and its practical implementation is limited by its footprint and a large number of SOAs.

SOA-based switches with a large switching fabric are not available commercially but they can be built using individual components described above. However, footprint and cost are the major concerns in building them in a large number of ports. A photonic integrated circuit (PIC) can be used to avoid these issues of SOA-based switches. For example, a monolithic integration of 16×16 port SOA switch has been demonstrated in [67, 68]. However, designing a large scale SOA-based switch using PICs is quite a challenging task. In the foreseeable future, integrated photonics can play a major role to realize these switches at feasible cost and footprint.

2.3 Evolution of Optical Interconnects for DCN

Optical interconnects have gained significant attention recently, as they seem to provide a promising and viable solution for future data centre networks compared to traditional electrical architectures. This section presents the optical interconnects schemes that have been proposed for data centre networks and provides a general insight on each architecture.

Optical interconnects are categorized into five categories according to the type of optical switches used: (1) architectures based on MEMS, (2) architectures based on SOAs, (3) architectures based on AWGRs, and (4) architectures based on WSSs, and (5) hybrid architectures based on fast and slow optical switches. A comparative analysis on the features of these schemes such as connectivity, scalability, cost and power consumption is also provided.

2.4 Architectures based on MEMS

2.4.1 Hybrid Electrical/Optical Switch

A hybrid electrical/optical switch (Helios) architecture for DCNs is proposed by [23]. Its architecture is shown in Figure 2.6. It consists of two level of switches called pod switches and core switches. The pod switch combines several ToR switches into a single unit and is interfaced to the core switches. Each pod switch typically holds between 250 and 1,000 servers. For simplicity, the ToR switch is considered as the pod switch here in Figure 2.6. The core switches are combination of electrical packet switches and MEMS switch which is also called optical circuit switch. Basic idea of this hybrid architecture is to use benefits of both electrical packet switches and MEMS switch. Bursty traffic between different pods is typically handled by the core electrical packet switches and long-lived, slowly changing inter-pod traffic is handled by the MEMS switch. The pod switch has a number of optical transceivers i.e. 10G Small

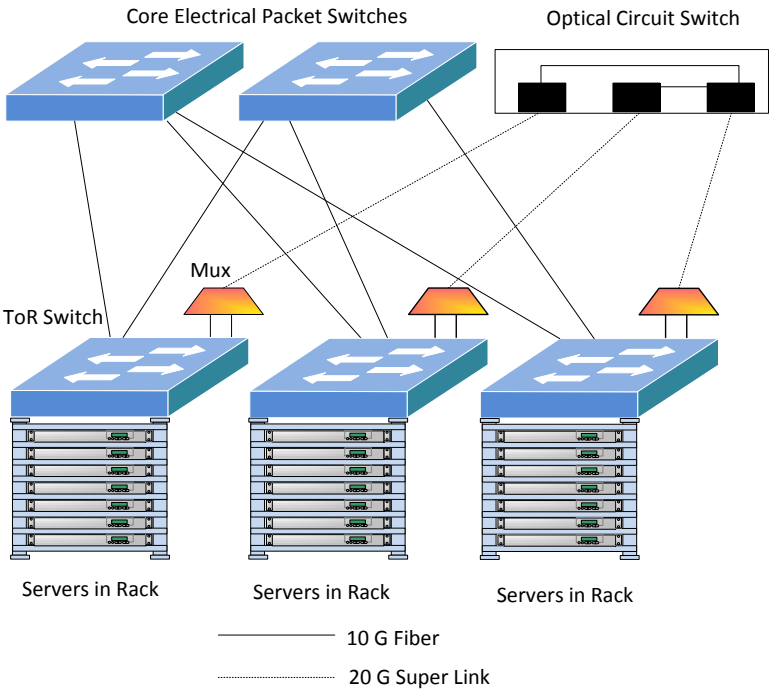


Figure 2.6. Helios Architecture.

Form Factor Pluggable Plus (SFP+) modules to connect to the core switches. Half of

these transceivers are connected to the core electrical switches. The other half of these transceivers pass through a multiplexer which multiplex these links into a single WDM link which is called a super-link. Super-link is then connected to the MEMS switch. In this way, 50% of bandwidth is shared between electrical and optical switches.

Control plane in Helios consists of three collaborating modules: a Circuit Switch Manager (CSM), a Topology Manager (TM), and a Pod Switch Manager (PSM). The CSM is the built-in controller of Optical MEMS which is responsible to configure optical MEMS switch. TM runs on server and is responsible for monitoring inter-pod traffic demands and calculates new configurations for optical switch. The PSM runs on each pod switch. It coordinates with TM and controls switch hardware and flow table. Layer 2 or 3 forwarding can be used for electrical packets switching.

Helios has shown reduction in cost upto a factor of 3, reduction in complexity upto a factor of 6 and reduction in power consumption upto a factor of 9 as compared to typical data centres using electrical packet switches only. It is also reported that if 50% of inter-pod traffic changes over multi second time-scales then Helios shows good performance by mixing of these switches. The other plus point of the Helios is usage of readily available optical/electrical components and reduced cost of upgrade. Higher data rates with MEMS switch can be achieved without any additional cost and complexity.

2.4.2 Hybrid Packet/Circuit Switch

A hybrid packet/circuit switch (HyPaC) [50] is also a hybrid optical/electrical interconnection architecture for data centres similar to Helios. Its architecture is shown in Figure 2.7. It consists of traditional electrical packet switches at different layers (access, aggregate and core) and a MEMS switch that is connected with all ToR switches. Servers are connected to the ToR switches and ToR switches are connected to the electrical switches at aggregate layer and a MEMS switch. Aggregate switches are connected to the electrical core switches. Each ToR switch is connected with exactly one ToR switch at a time by using optical network and this reconfiguration changes very slowly with the time. Reconfiguration is based upon traffic demands between racks.

Pair of racks demanding high traffic communication are connected through the MEMS switch. The circuit switch network can only offer a matching on the graph of racks and Edmonds algorithm [69] is used to compute matching between racks. HyPaC is

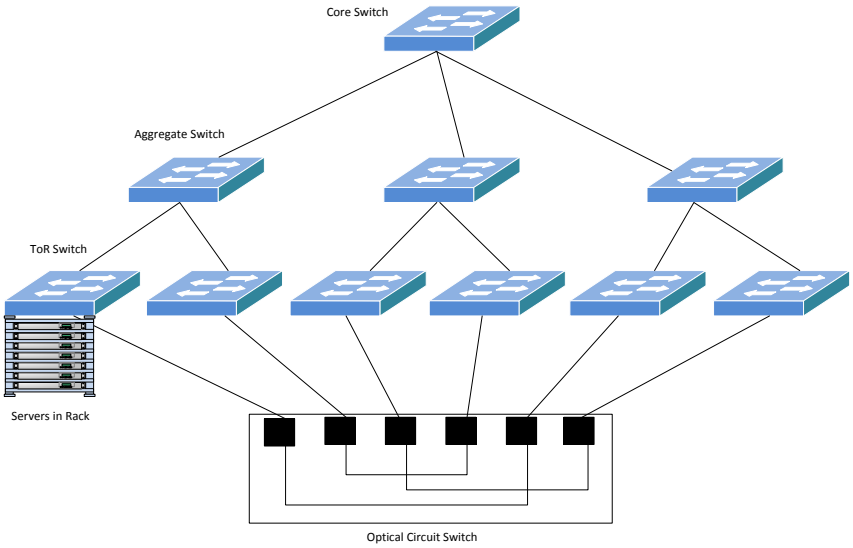


Figure 2.7. HyPaC Architecture.

different from Helios in that the end hosts are responsible for traffic demand estimation while in Helios topology manager is responsible for this functions. Traffic demand estimation is done by increasing the size of per-connection socket buffer and observing end-host buffer occupancy at runtime. It needs extra kernel memory for buffering, but is transparent to applications and does not require switch changes. HyPaC separates the two networks by using VLAN based routing mechanism. End host tags each packet for specific VLAN and ToR switches are responsible for forwarding packet either to the optical switch or electrical switch based on the tag done by end host. Each host runs a management daemon that informs the kernel about the inter-rack connectivity. The kernel then de-multiplexes traffic to the optical and electrical paths properly. The optical configuration manager runs on the server and collects traffic measurements, determines how optical paths should be configured, issues configuration directives to the switches, and informs hosts which paths are optically connected.

HyPaC is suitable for certain types of applications which require huge data transfer components e.g. VM migration or backup of servers, skewed traffic patterns, and loose synchronization.

Due to the large switching time of MEMS switch and the control plane overhead required for the estimation of traffic demand and the calculation of a new OCS topology, the control plane has the limitation to support applications that have high traffic stability, i.e. workloads that last several seconds. So Helios and HyPaC are suitable in scenarios where traffic stability is high. Power consumption is another limitation in these techniques due to the usage of electrical switches and power hungry transceivers at the core which is not desirable in future large scale data centres. Helios and HyPaC architectures are also not scalable because their scalability is limited by the constraint of the number of optical ports of the MEMS switch.

The performance degradation for hotspots (ToR pairs with high traffic demand) is another major limitation in these techniques. For example, if traffic demand of hotspots is higher than the maximum supported bandwidth in OCS-MEMS path then the performance bottleneck arises. Other MEMS based interconnects such as optical switch architecture [54, 70], reconfigurable architecture [71, 72] and hybrid reconfigurable architecture [53] as describe in the next sections do not have this limitation because they provide full bandwidth between any ToR pair.

2.4.3 Optical Switch Architecture

The optical switch architecture (OSA) for DCNs [54, 70] comprises a two layer architecture i.e. edge and core. At the edge, it consists of ToR switches while at the core, it consists of a MEMS switch. Its architecture is shown in Figure 2.8. Between MEMS and ToR switches, it uses a wavelength selective switch in combination with multiplexer/demultiplexer, coupler and circulator at the core. Servers are connected with ToR switches. ToR switches have N number of distinct wavelength transceivers. These transceivers use distinct fibres to connect to send and receive infrastructures. On the sending fibre links, these are multiplexed into one single fibre link and are divided by k set of wavelengths using WSS where $k < N$ and each of the k group is transmitted on its own fibre by the WSS. On the receiving end, these k groups are first coupled into single fibre by using coupler and then are demultiplexed into different wavelengths where they are then connected to the receivers.

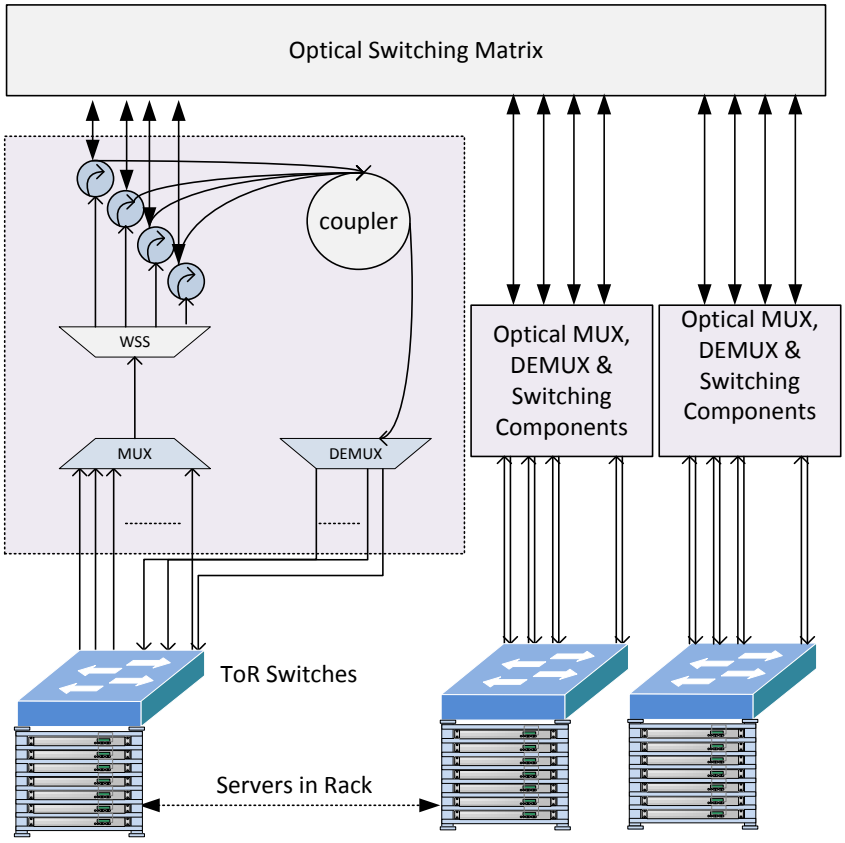


Figure 2.8. OSA Architecture.

The main theme of the OSA is to connect ToR switches through the MEMS switch by a single hop that generate high inter-rack traffic, while bursty and low volume traffic are assigned to multi-hop connections. For example, ToR A has a large amount of traffic to send to the ToR B and ToR B has a large amount of traffic to send to the ToR C. ToR A also has a small amount of traffic to send to the ToR C. So in this case, ToR B is connected to the ToR A and C directly through the MEMS switch by a single hop and traffic from ToR A to C is first sent to the ToR B which forwards the traffic to the ToR C. The intermediate ToR switches in multi-hop path receive the packet and convert it into an electrical signal for processing, and then regenerate the packet into optical domain by optical transceiver for destination ToR switch. In this way, low volume traffic can take more than one hop to reach its final destination.

The decisions of MEMS switch configuration are made by topology manager (TM) which is connected with MEMS, WSS and ToR switches. The TM gets traffic matrix

from the ToR switches, finds suitable configurations, and applies them to the MEMS, WSS, and ToRs. In OSA, each ToR can connect with k ToR switches simultaneously and therefore the degree of OSA is $k > 1$. The configuration of the MEMS determines which set of ToR switches is connected. Flexible link capacity can be achieved with WSS by grouping more wavelengths into a single group for the ToR switches when traffic demand is high.

The main advantage of the OSA is that it is based on commercially available optical components and is power efficient as compared to Helios and HyPaC. Like Helios and HyPaC, the OSA is also well suited for applications in which connection lasts more than a second while bursty and short duration traffic can still take advantage of multi-hoping to reach final destination with additional cost of end-to-end delay. The OSA is more flexible than Helios and HyPaC because it provides full bisection bandwidth between any ToR pair.

The OSA does not perform well in situations where most of the traffic flows last few milliseconds. The multi-hoping technique not only increases latency but also increases power consumption of ToR switches. Similar to Helios and HyPaC, the scalability of the OSA is also limited by the constraint of the number of optical ports of the MEMS switch. The reconfigurable Architecture [71, 72] as described in the next section does not has the limitation of scalability.

2.4.4 Reconfigurable Architecture

The reconfigurable architecture has a two layer topology [71, 72]. It design is shown in Figure 2.9. It relies on a single stage shuffle exchange topology having electronic packet switches (ToR) at the edge and array of MEMS switches at the core. Each ToR switch has k switching elements (SEs) which are attached to the optical core. It means, each ToR switch is connected with the maximum of k other ToR switches at a time. Optical core is logically divided into different planes. This scheme also uses multi-hopping technique similar to OSA. The idea to segregate traffic volumes is similar to the OSA i.e. the high traffic demanding ToR switches are connected with a single hop while low and distributed traffic have to travel multiple hops in order to

reach final destination. A packet needs to traverse at least one hop and at most $log_k N$ hops to reach its destination. ToR switches are programmed so that they can forward the packet based on destination addresses. The tag based routing is used for routing the packets travelling on multi-hops. The tag is generated according to the number of hops. Large number of hops have the larger tags and small number of hops have the smaller tags associated with the packets. The packet is forwarded based on the most significant bit in the tag.

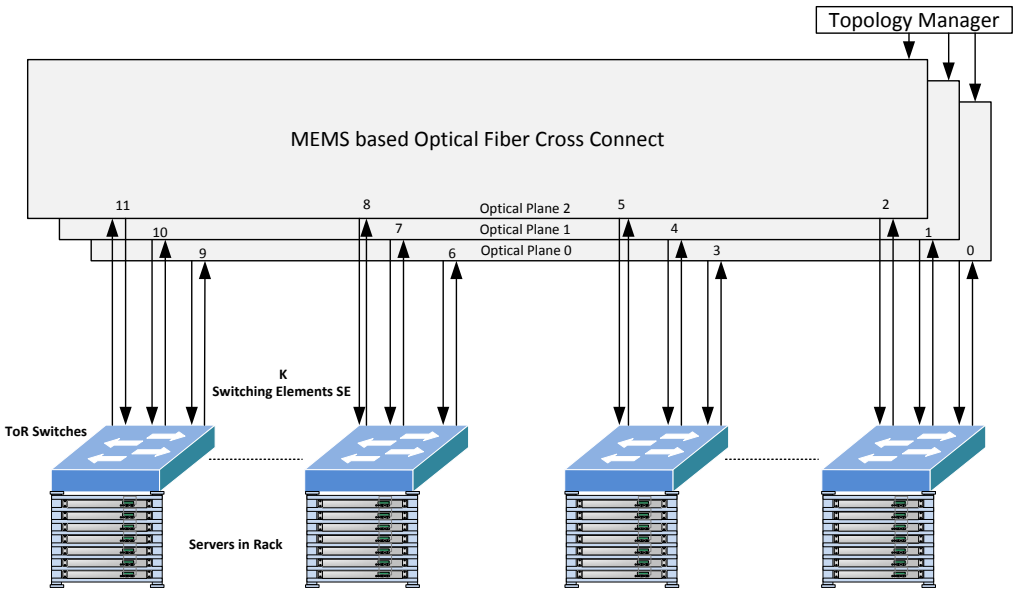


Figure 2.9. Reconfigurable Architecture.

Topology Manger (TM) is responsible for the configuration of optical core plane. The TM is responsible for adjusting the connectivity of the shuffle to match physical interconnection to dynamic logical topology.

The main advantage of this scheme is its implementation simplicity because it is based on readily available optical components. Unlike OSA, this architecture is scalable due to the usage of single stage shuffle exchange topology with multi-hopping. Like other MEMS based interconnects, this scheme is suitable for long lived traffic flows. The lowest cost of upgrade is another plus point of this architecture. However, this scheme is not suitable if majority of the traffic is bursty and short-lived because it

will increase latency and power consumption due to multi-hopping technique. A hybrid reconfigurable architecture [53] as described in the next section combines benefits of Helios, HyPaC, OSA and reconfigurable architecture into a single approach.

2.4.5 Hybrid Reconfigurable Architecture

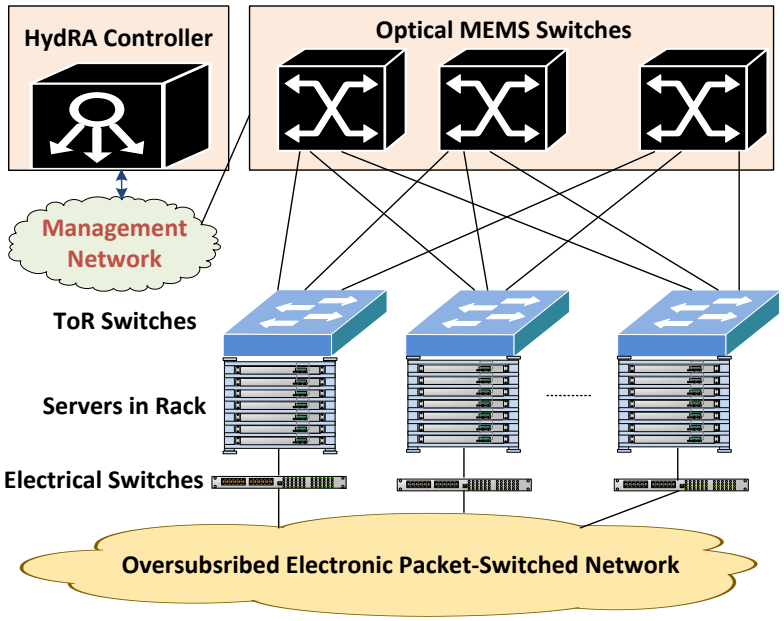


Figure 2.10. HydRA Architecture.

The hybrid reconfigurable architecture (HydRA) is another hybrid optical/electrical data centre network proposed by researchers in [53]. Its architecture is shown in Figure 2.10. The HydRA combines approaches used by Helios, OSA and Reconfigurable architecture into a single approach. The ToR switches are used at the edge and array of electrical switches and optical MEMS switches are used at the core while in Helios and OSA, only one optical MEMS switch is used at the core. The array of multiple optical MEMS switches at the core was proposed in Reconfigurable architecture. The multi-hopping technique is used with optical MEMS switches for long-lived traffic flows while short-lived or traffic during MEMS switches reconfiguration is diverted through electrical switches at the core. The control plane comprises HydRA network controller that is responsible to compute efficient workload-input specific topologies, ToR switches port tagging and MEMS switches configuration. The Hy-

dRA controller also comprises Floodlight OpenFlow controller that pushes the rules to the OpenVswitch running at the servers in the rack.

In [53], the researchers also demonstrate a prototype implementation of the HydRA network that uses Ethernet switches and a custom-built software controller with real applications. The HydRA features advantages of all other MEMS based optical interconnects. However, like all other MEMS based architectures, the control plane of the HyDRA also has the limitation to support applications that have high traffic stability, i.e. workloads that last several seconds. This limitation is due to the large switching time of the MEMS switch and the control plane overhead required for the estimation of traffic demand and the calculation of a new OCS topology.

In the next section, another types of architectures that are based on SOAs are presented.

2.5 Architectures based on SOAs

2.5.1 Optical Shared Memory Supercomputer Interconnect System

Optical shared memory supercomputer interconnect system (OSMOSIS) architecture proposed by IBM [73–75] is shown in Figure 2.11. It is based on a broadcast-and-select configuration, and provides full-duplex connection at 40 Gbps data rate. It is a 64 port optical packet switch which introduces cell switching of fixed sized packets of 256 bytes known as cells. Electronic control logic is implemented by using field-programmable gate array (FPGA) to control scheduling of cells. Switching is performed by reconfiguring the optical core on a cell-by-cell basis. There are two basic modules i.e. broadcast and select modules. Broadcast modules contain transmitters, multiplexers, amplifiers and splitters while select modules consist of SOAs, multiplexers, demultiplexers, and receivers. Packets are buffered at the ingress of the ToR switch and are multiplexed onto a single fibre using multiplexer. Eight distinct wavelengths are multiplexed in each broadcast module and after amplifying the signal

from amplifier, the signal is broadcast by using 1 × 128 power splitter. There are two stages in the select module of this architecture. Two stages of eight SOAs each are used as gate elements by utilizing their on/off behaviour. First stage of SOA is used to select particular fibre of desired signal output and second stage of SOA is used to select particular wavelength of desired output receiver. The centralized controller is responsible for switching on and off behaviour of these SOAs.

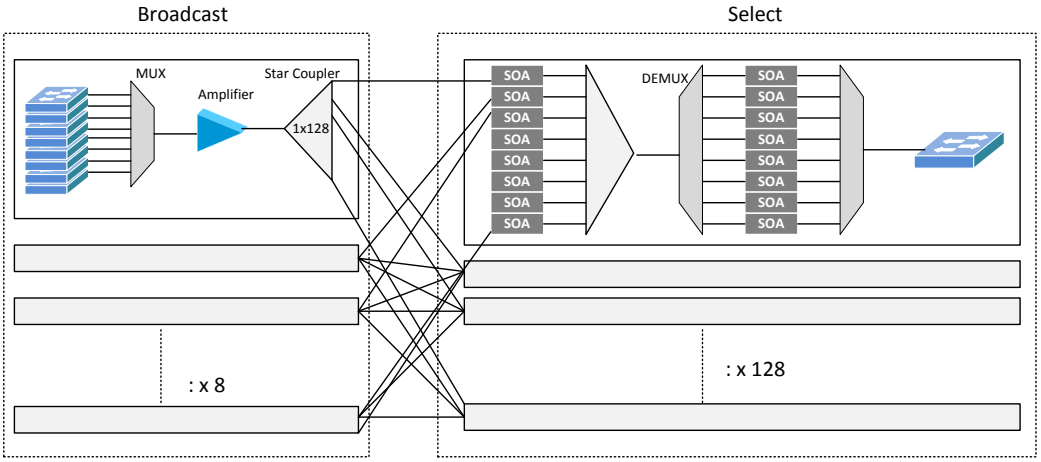


Figure 2.11. Osmosis Architecture.

FLPPR (Fast Low-latency Parallel Pipelined arbitration) crossbar scheduler algorithm [76] is developed that provides a parallel implementation in FPGAs and achieves 51.2-ns packet cycle time for 64 ports. Speculative transmission mechanism is used to achieve minimum latency for request - grant - transmit cycles. For speculative transmission, two paths from each input to a given output were used in broadcast-and-select architecture resulting a 64 × 128 crossbar switch. To achieve zero packet loss, a flow-control mechanism was also adopted to prevent packet buffer overruns.

Major advantage of this scheme is the low latency of packet transmission. There are many drawbacks in this scheme. First, it is a costly design in terms of capital expenditure (CAPEX) because it uses two receiver for every output port and for each receiver, it uses 16 SOAs which are expensive devices. Second, this scheme is not scalable to build thousands of nodes of interconnect due to topology of the architecture and complexity of the centralized controller. Data vortex [5,77] is the next SOA-based

architecture which targets issue of scalability. It is described below.

2.5.2 Data Vortex

Data vortex is a distributed interconnection network [5, 77]. Its architecture is shown in Figure 2.12(a). It uses optical packet switching and is based on SOAs which are used as a gate switching element similar to the OSMOSIS. Its network topology is based on a banyan structure and incorporates a distributed deflection routing scheme which removes packet contention without the usage of optical buffers. The nodes are arranged in cylinder C, with height h and angle A as shown in Figure 2.12(b). Each cylinder corresponds to 1 stage of the banyan network. Each node in a cylinder consists of 2×2 switching elements (SOAs) which are arranged in a fully connected directed graph. A 2×2 SOA switch uses four couplers and is controlled by controller which controls ON/OFF switching of SOAs. FDLs are used to temporarily delaying packets during switching. Packet is routed to deflection route in case of congestion in the network. Straight lines depict injection fibres, curved lines show deflection fibres and dotted lines show control cables. Each 2×2 switching element has 2 input fibres each from north and west and two output fibres each for south and east. The prototype design of the data vortex is presented recently [78].

The biggest advantage of this scheme is the scalability as it can be scaled to the thousands of nodes using modular architecture and SOAs can also recover power losses which occur due to multi-stage path of a packet. The major drawback of the data vortex is high end-to-end packet latency due to multi-stage architecture. FDLs can only provide limited delay and packet losses can also occur due to congestion in both desired and deflection route which is not feasible in the data centres. Bidirectional optical interconnect network for data centres [79] is another scalable architecture based on SOAs. Unlike to all other SOA-based architectures, bidirectional architecture provides bidirectional communication. It is described below.

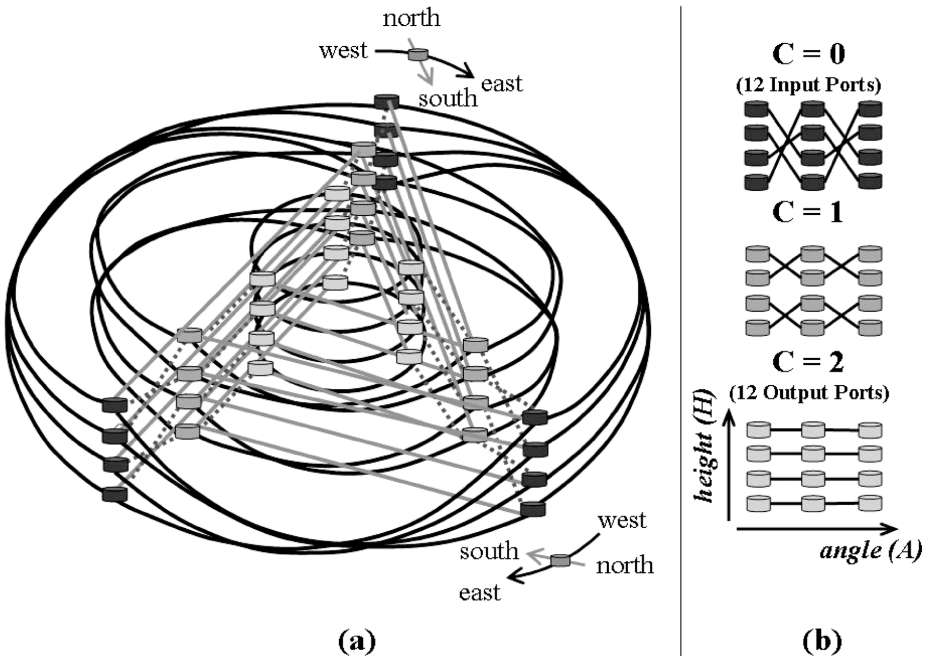


Figure 2.12. Data Vortex Architecture [5].

2.5.3 Bidirectional Architecture

Bidirectional optical interconnect network for data centres [79] is based on bidirectional 2×2 SOA switch [80]. 6 SOAs are required to build 2×2 switch. Bidirectional architecture is shown in Figure 2.13.

Prototype of 2×2 bidirectional switches has been developed using (2-ary 3-tree) Banyan network (k-ary n-trees) topology [80]. It can support k^n processing nodes with n stages of $k^{n-1}k \times k$ switches. 2×2 bidirectional switch can also be implemented using broadcast-and-select configuration but it requires 4×4 switch. Number of SOAs required would be 16 in this case and its cost would scale as k^2. 6 SOAs are required to build 2×2 switch using proposed prototype. It uses 4 nodes at 40 Gb/s (10×4) speed supporting 4 wavelengths ranging from 1546.12 nm to 1548.52 nm using 4 DFBs. Bit error rate of 10^{-12} was observed. Scalability is the major advantage of this scheme.

Implementation complexity, control mechanism and CAPEX cost are the major challenges in this scheme. Space wavelength architecture [29] is another SOA-based architecture which targets low latency and high scalability. It is described below.

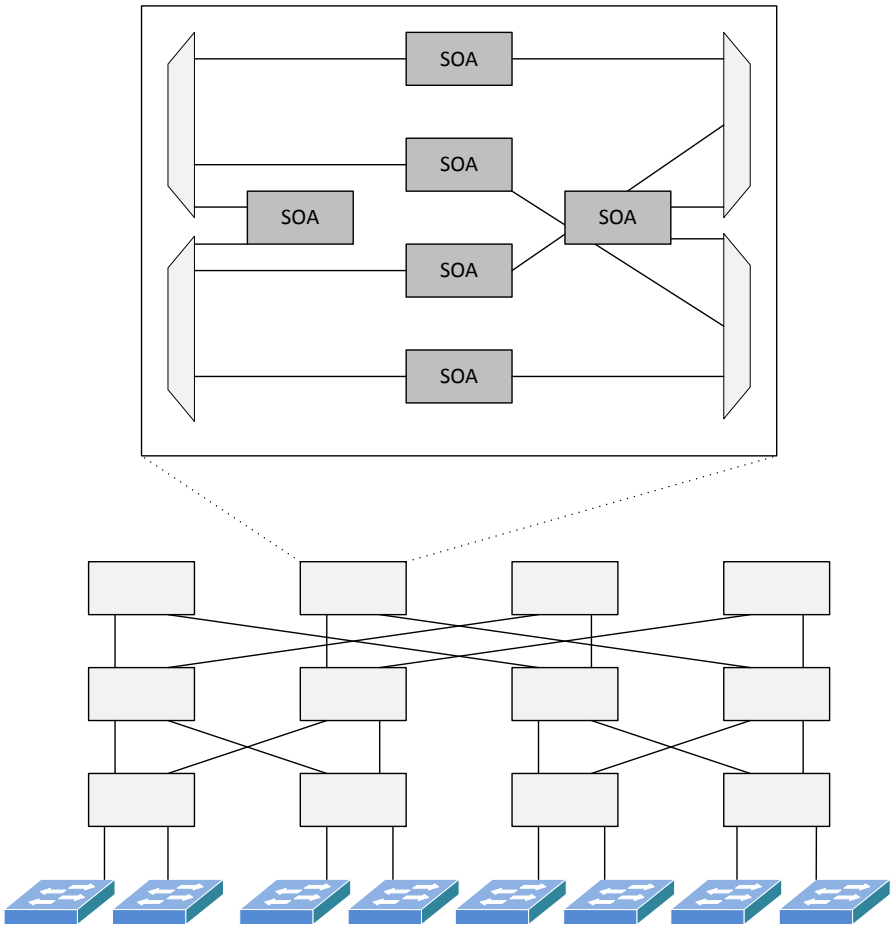


Figure 2.13. Bidirectional Architecture for DCNs.

2.5.4 Space Wavelength Architecture

Space wavelength (SW) interconnection architecture [29] utilizes benefits of both wavelength and space domain, and is shown in Figure 2.14. It consists of M independent cards. Each card has N inputs and $N \times M$ output ports at the sender side and $M \times N$ input ports at the receiver side. Each card has intra-card scheduler which contains N input queues for each input port and buffers packets temporarily before transmission. Each card also has an array of N fixed lasers operating at N different wavelengths connected with modulator. Modulated input signal is then broadcast from $(1 \times M)$ power splitter on particular wavelength to $(M : 1)$ SOA switch fabric. Each receiver is connected with M SOAs which act as ON/OFF switch similar to OSMOSIS. After selection, signal is passed to the receiver. This modulated wavelength is

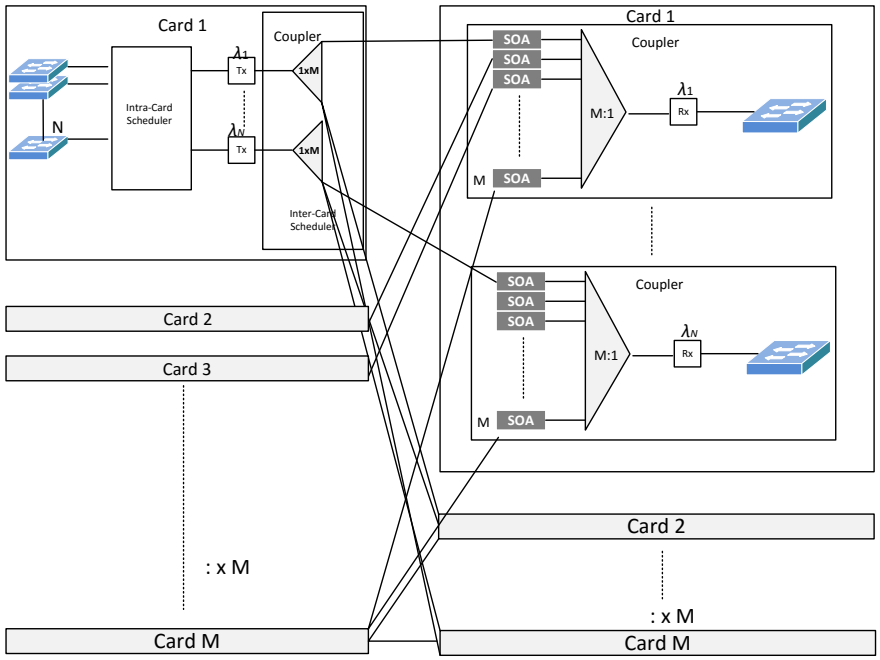


Figure 2.14. Space Wavelength Architecture.

controlled by the inter-card scheduler. In order to route a packet from an input port to an output port, first the destination card is selected by setting the $(1 \times M)$ space switch and then destination port is selected by selecting the wavelength to be modulated. Every card has N distinct wavelengths attached to the receivers. This scheme provides low latency and can be scaled by adding more wavelengths.

The major drawback of this scheme is the CAPEX cost. M SOAs per single output port are required to switch packet and this cost increases linearly as the size of the switch increases which makes it a costly solution. Space time interconnection architecture [81] is an advanced version of SW architecture. It is described below.

2.5.5 Space Time Interconnection Architecture

Space time interconnection architecture (STIA) [81] utilizes space, wavelength and time domain unlike to SW architecture which only employs space and wavelength domain. The wavelength domain is used to increase throughput, the space domain to switch packets between different cards and the time domain to switch packets between

different ports in card. Block diagram of this interconnect is shown in Figure 2.15. It consists of M cards and each card has N inputs and M output ports.

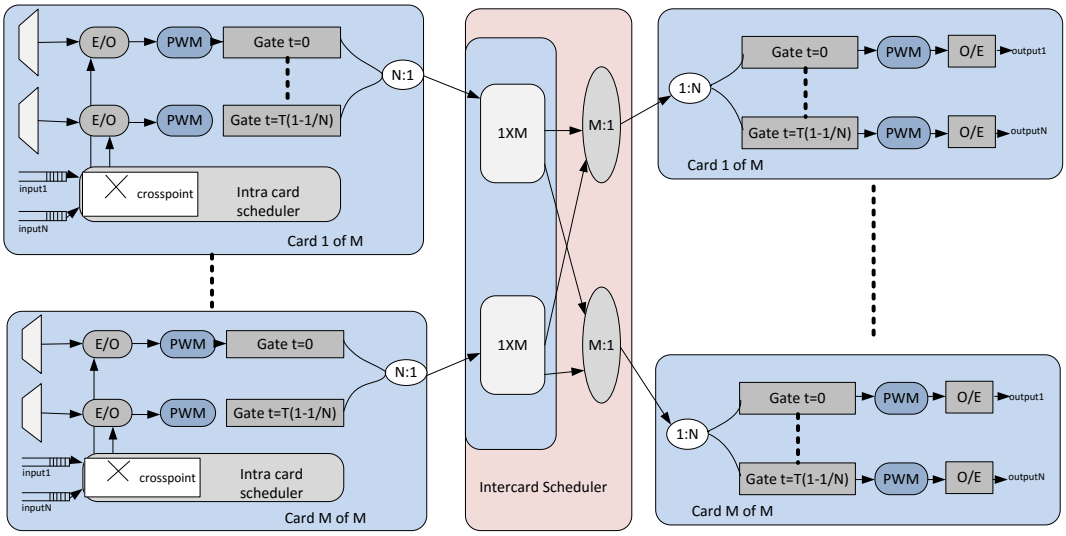


Figure 2.15. Space Time Interconnection Architecture.

The optical packet is generated from serial electrical packets of duration T and is modulated with combination of N optical channels with a single broadband modulator. Packet is then sent to passive wavelength stripped mapping (PWM) which delays each channel by $\frac{T}{N}$ and delayed channels are gated to get packet of duration $\frac{T}{N}$. In this way, packet is compressed in time by the number of wavelength channels N. Mach-Zehnder with combination of N array lasers are used for packet generation in WDM channels. N copies of the packet of duration T is generated and is compressed by PWM and SOA gate to get packet of duration $\frac{T}{N}$. The $\frac{T}{N}$ duration of time slot is used for every packet and is dedicated to each output port. $N:1$ coupler is used to combine signal from every input channel each having $\frac{T}{N}$ duration and is sent to $1 \times M$ space switch based on SOA similar to SW architecture. There are two types of scheduler: intra-card and inter-card similar to SW architecture. Intra-card scheduler is responsible for scheduling decision of packets and inter-card scheduler is used to control $M:1$ space switch. On the receiver side, $1:N$ splitter broadcasts the packets to each output port. SOA gates select proper time slot and PWM again delays the packet of duration $\frac{T}{N}$ to get packet of duration T in reverse direction and receiver converts packet into electrical domain.

Similar to SW architecture, STIA also provides low latency and high throughput. STIA can be scaled efficiently and is recently shown in latest work [82] by arranging STIA in tree, folded clos or flattened butterfly topology but in every upper level, packets have to be converted into electrical domain and all the processes of packet generation, scheduling and sending are started again from scratch in the next level.

It is however a costly solution due to the usage of broadcast and select architecture of $M \times M$ space switch. Folded clos or flattened butterfly topology of STIA not only increases end-to-end packet latency but also energy consumption due to O-E-O operation.

In the next section, architectures based on AWGRs are presented.

2.6 Architectures based on AWGRs

2.6.1 Low-latency Interconnect Optical Network Switch

Low-latency interconnect optical network switch (LIONS) architecture has been proposed in 2013 [27]. It was previously proposed under the name DOS (Datacenter Optical Switch) in 2010 [83]. The LIONS architecture aims to overcome the limitations of buffer scalability in DOS. The block diagram of DOS/LIONS is shown in Figure 2.16. Switching fabric is based on AWGR which is used in combination with tunable wavelength converters (TWCs). It is possible to route different inputs to a same output simultaneously by using AWGR's cyclic wavelength routing technique. It also provides fully connected topology and non-blocking characteristics because each output port is connected to each input on distinct wavelength. TWCs are used to tune wavelengths at each input according to the wavelength of the output. Nodes are connected with optical channel adapters (OCA) that are used to generate optical packets in conjunction with optical labels. Label switching is used to route packets from inputs to outputs in which labels are sent on different wavelengths [84]. The packet is sent to the label extractor (LE) which extracts label from it by using filter and sends the label to the control plane for routing decision. FDLs are used to store optical payload

temporarily until the routing decision of the control plane arrives. The control plane processes label and sends appropriate wavelength request to TWC to tune on specific wavelength. TWCs can be configured in nanoseconds scale [27]. The packet is then sent to the AWGR and is reached to the desired output. It can be seen that each output can take N number of inputs on different wavelengths simultaneously. N receivers can be connected with each output, and by using demultiplexing, particular receiver receives packet on specific wavelength after converting it into electrical domain.

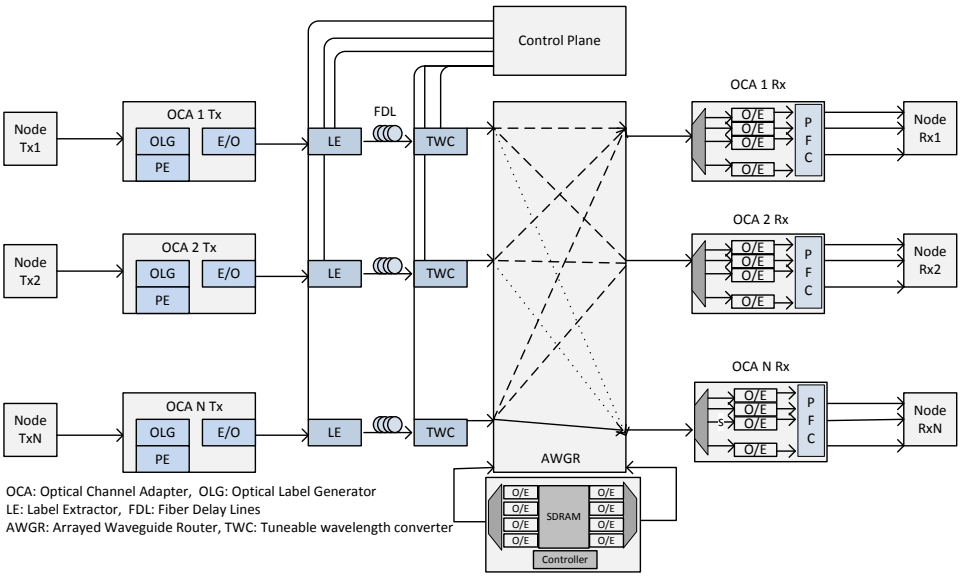


Figure 2.16. DOS/LIONS Architecture.

Loopback shared buffer is attached with the AWGR which is used to store packets temporarily in case of output port contention or if decision is not made by the control plane within specific FDL delay. In DOS, SDRAM is used to store packets in shared buffer which converts packets to electrical domain. The request is sent to the control plane to reschedule the packet and waits until a grant is received. After receiving the grant, the packet is removed from the shared buffer, it is converted into optical domain and is sent to the AWGR by tuning it into specific wavelength by TWC. LIONS uses three different loopback buffering schemes: 1) Shared loopback buffer (SLB) which was also used in DOS; 2) Distributed loopback buffer (DLB); and 3) Mixed loopback buffer (MLB). DLB uses N separate memory units realizing N queues as compared to single SLB in DOS. It also occupies N input and output ports while MLB occupies only

one output port and is demultiplexed to N ports in loopback buffer. A 40 Gbps (8×8) prototype of the DOS architecture has been demonstrated by Roberto [85].

In both DOS and LIONS, 2-phase arbiter is used in the control plane for packet scheduling. Each input sends request to the arbiter in the control plane and waits for the grant. The number of inputs competing for a given output in the worst case is decreased by a factor k, where k is the number of wavelengths allowed per AWGR output. If $k = N$ then no arbitration is required because there is no contention, but typically $k < N$, so there is contention, but only the inputs in the corresponding contention group need to be examined by the arbiter to grant a request. It is not necessary to look at all the inputs [27].

Scalability of a single stage DOS depends upon the scalability of the AWGR, capacity of shared loopback buffer and the capability of TWCs. Major drawback of loopback buffer in DOS is its capacity which has been eliminated by using DLB and MLB buffers in LIONS but it still requires costly O-E-O conversion. This loopback buffer is then eliminated in another version of LIONS named as NACK-LIONS [86] by using all optical negative acknowledgement technique. In this technique, $(N+1) \times (N+1)$ configuration of AWGR is used. Instead of storing packet in loopback buffer, the packet is sent to the $N+1$ reflective port which reflects the packet back to the sender by using optical circulator. The dedicated receiver is placed in the sending host which receives the packet coming back to the sender. This packet acts as AO-NACK. The host then resends the packet. NACK-LIONS is also based on the centralized control plane due to which the scalability of this architecture is also limited by the control plane.

Another variation of LIONS with the name TOKEN-LIONS has been proposed in [87]. TOKEN-LIONS uses distributed control plane instead of the centralized controller. This distributed control plane does not have global information. TOKEN-LIONS is based on exploiting the saturation effect of reflective semiconductor amplifiers (RSOAs). RSOA is deployed at each output port of AWGR and is used as mutual exclusion type of arbiter. Gain saturation effect of RSOA is used to realize mutual exclusion behaviour. Packets are buffered at end hosts instead of input ports. The control requests are sent ahead of the data packets. The request saturates RSOA and

the power is reflected back to the sender which is considered as a grant. After receiving the grant, the packet is sent. On the other hand, if another request comes and previous request is in process, it means RSOA is already saturated and half of the power is reflected back which is considered as a decline at the end host and the request is excluded. The major advantage of the TOKEN-LIONS technique is the scalability of the control plane due to the distributed control plane which does not require global information. However, the delay at input host can negatively effect overall performance due to waiting time of token from RSOAs.

Major advantage in these schemes is low latency because TWCs are fast and provide configuration within nanosecond range. Although, the power consumption and scalability of loopback buffer has been eliminated in LIONS, but the scalability of overall architecture still depends upon the number of AWGR ports. Hi-LION is another version of LION which is proposed recently to address scalability issue [88]. Hi-LION is a hybrid architecture that employs OPS and electronic packet switching (EPS) using AWGR and electrical switches. This architecture is scalable to hundred thousand nodes. The CAPEX cost of the interconnect is still a major concern because TWCs are expensive devices and are required for each input port of AWGR. Also, multiple receivers per output port is a costly solution in terms of CAPEX cost.

2.6.2 TONAK-LION

TONAK-LION [89] is the advanced version of LIONS and all of its related architecture. It utilizes the benefits of both NACK-LION and TOKEN-LION architectures. A block diagram of this architecture is shown in Figure 2.17. It consists of two $(N \times N)$ configuration of AWGR for N input ports each. One AWGR is used for the control plane and another one is used for the data plane. Optical channel adapter (OCA) is installed at each host and consists of a tunable laser (TL) which generates both packets and token requests (TRs). The OCA is also equipped with Mach-Zehnder, circulator, splitter and negative acknowledgement detector(ND). A line card is placed at each input port which is connected to both AWGR using 1:2 switch each for the control and the data plane respectively. It is also equipped with circulator, FPGA controller, token detector

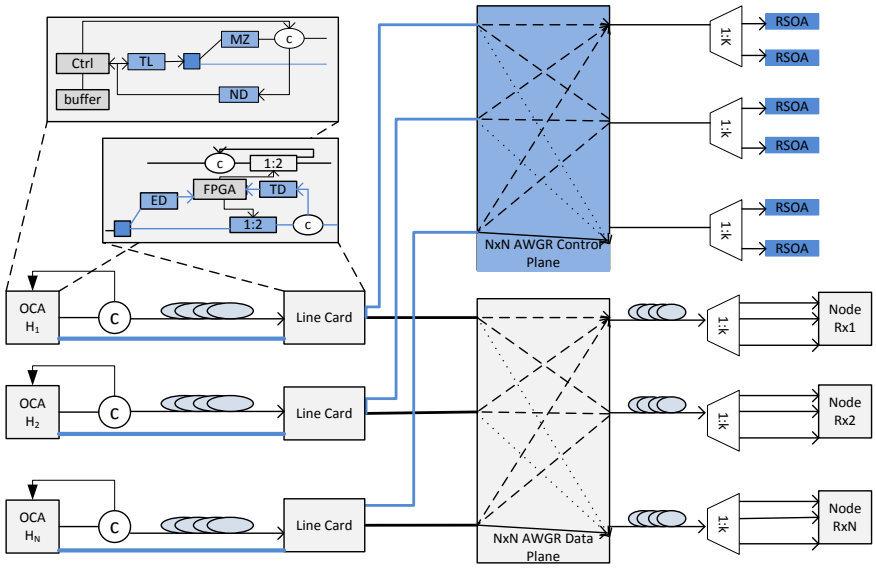


Figure 2.17. TONAK-LIONS Architecture.

(TD) and edge detector (ED). In the control plane, k RSOAs are deployed at each output port and in the data plane, k receivers are deployed at each output port so that k parallel channels simultaneously reach at each output port. The TRs are generated before a packet by the tunable laser at a particular time which is called offset time. This offset time is equal to the round trip time from the line-card to the RSOA. The packet is released after the offset time from the OCA and is delayed in FDLs ahead of the line card. The TR from the OCA is reached at the line card and the line card sends it to AWGR by configuring 1:2 switch at optical plane. The TR is received by one of the desired RSOA. The RSOA amplifies this request and sends it back to the AWGR input port where it is received by the circulator and is sent to the token detector (TD). The TD converts TRs into electrical domain and generates an electrical signal of voltage proportional to the power of the received TR. If the voltage is greater than the send voltage, then it means port is available. FPGA controller sets the 1:2 switch in the data path to cross state so that packet can be transmitted to the data plane. In case of contention, the RSOA is already saturated and amplifies the TR with low power and sends it back to the input port of AWGR. The TD generates electrical signal based on the received power and FPGA then sense that the output port is not available due to the difference of power level. The controller then sets the 1:2 switch in the bar state

and the packet is sent back to the OCA for retransmission. The ND placed at OCA detects packet and it is considered as negative acknowledgement and this process starts again from beginning.

The major disadvantage with this scheme is the CAPEX costs and implementation complexity. 2 AWGR are used for each input port and k RSOAs for each output port makes it a costly solution. The architectures based on AWGRs described above are based on a single stage topology and their scalability is limited by the number of ports of AWGRs and tuning range of TLs/TWCs. In the next section, Petabit optical switch [90] is presented which is also based on AWGRs but it uses three stage topology to address the limitation of scalability.

2.6.3 Petabit

Petabit optical switch [90] is shown in Figure 2.18. It is also based on AWGR in conjunction with TWCs. A three stage Clos-network is used and the stages are called as input modules (IMs), central modules (CMs) and output modules (OMs). Unlike DOS, Petabit is a buffer-less optical switch using 3 stages of AWGR and buffering is only done at the line cards connected with ToR switches. Each input port of the CMs and OMs has a TWC that can be used to control the routing path while IMs do not have TWCs because the wavelength at their input ports are tuned by tunable laser source on the line cards.

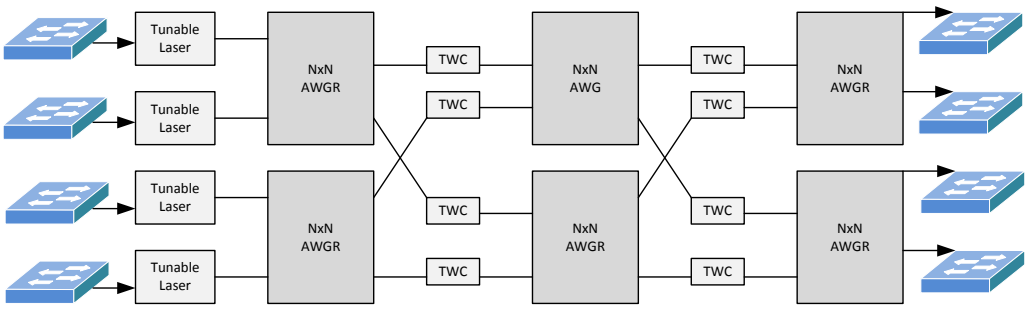


Figure 2.18. Petabit Architecture.

Contention management is done using electronic buffers in the line cards and an

efficient scheduling algorithm. Virtual output queues (VOQs) are used to store packets temporarily in the line cards. Each VOQ is maintained per OM instead of per output port which reduces the number of VOQs and complexity of the algorithm. Frame based switching is employed in which packets are assembled into fixed size frames (200 ns) in ingress of the line cards and are released at the egress of the line cards.

Major advantages of petabit are scalability, high throughput and reduced latency. The major drawback is high CAPEX cost because TWCs are expensive devices and this cost has been increased 3 times as compared to LIONS by using TWCs in 3 stages of the switching fabric. The other limitation is the implementation complexity of the controller and the scheduling algorithm. In the next section, an integrated router interconnected spectrally project is presented which is also based on a three stage topology but it uses a combination of space and time switches.

2.6.4 Integrated Router Interconnected Spectrally Project

The Integrated router interconnected spectrally (IRIS) project [91, 92] is also based on a three stage architecture and is shown in Figure 2.19. The first stage is a space switch that sends packets evenly to the ports of the second stage. The second stage is a time switch that works in a round robin fashion and it stores packets by adding different delay and delivers the packets to the third stage which is also a space switch connected with the output fibres.

Space switches consists of $N \times N$ AWGR and an array of wavelength switches (WSs). The WS demultiplexes each input to different wavelength and uses wavelength converters (WCs) to convert each wavelength to a specific wavelength and then multiplexes all the wavelength. Time switch consists of an array of time buffer (TBs). Each TB can store N concurrent packets by multiples of the packet-slot duration using FDLs. It can also drop packets in case of buffer overflows.

A prototype of IRIS has also been demonstrated in integrated optical chips by using FPGA based card which converts four 10 Gb Ethernet streams into one bidirectional 40 Gb/s IRIS packet stream by using 4 XFPs and 4 pairs of 10 Gb/s multiplexers and

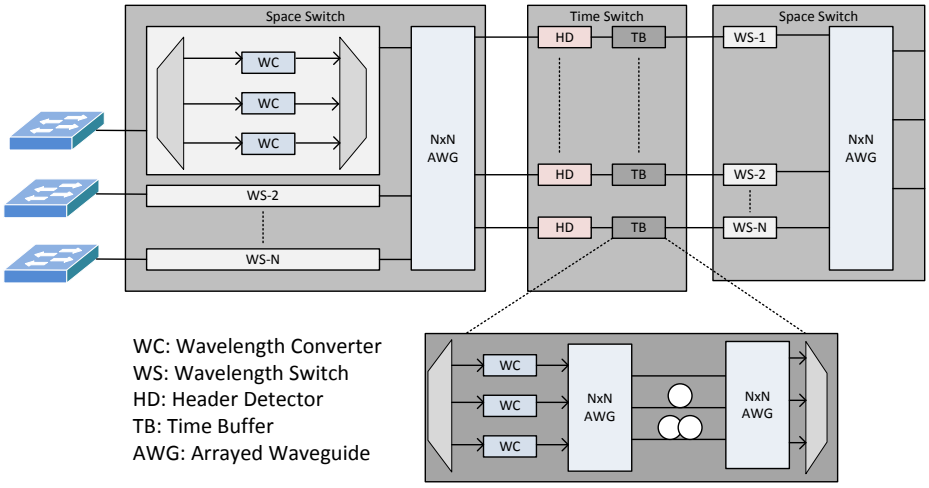


Figure 2.19. IRIS Architecture.

de-multiplexers. The 40 Gb/s wavelength converter is a fully-integrated InP circuit that uses an SOA for wavelength conversion. It can be done internally by using integrated fast-tunable multi-frequency laser (MFL) or externally by a sampled-grating distributed Bragg reflector (SG-DBR) laser. Optical switch in this example is based on a passive silica chip with dual 40×40 AWGRs.

This is the most expensive design in terms of CAPEX cost as compared to other AWGR based interconnects. The architectures based on AWGRs described above use TL/TWCs as a source of input into AWGRs. In the next section, an orthogonal frequency division multiplexing (OFDM) technology based optical interconnect is presented that does not require TLs/TWCs.

2.6.5 OFDM-based

Optical interconnect based on AWGR and multiple-input multiple-output (MIMO) OFDM technology with parallel signal detection (PSD) is presented in [93]. The block diagram is shown in Figure 2.20. Each ToR switch is equipped with OFDM modulator which receives aggregated signals and modulates these signals into k OFDM data signals with proper sub-carrier. K represents number of destination racks. O-OFDM technique is based on an OFDM signal generated electrically and modulated to an optical

carrier. These data signals are converted to k O-OFDM optical signals through directly modulated laser which are then combined with WDM combiner to form OFDM signal then it is sent to $N \times N$ AWGR. Signals are demultiplexed into input port of AWGR and different signals are sent to different output ports by using different wavelengths. Signals are then combined at each output port of AWGR and are sent to the receiver. In this way, each ToR switch can send the same OFDM signal to many destination racks at the same time, and many ToR switches can send the signal to the same destination rack simultaneously.

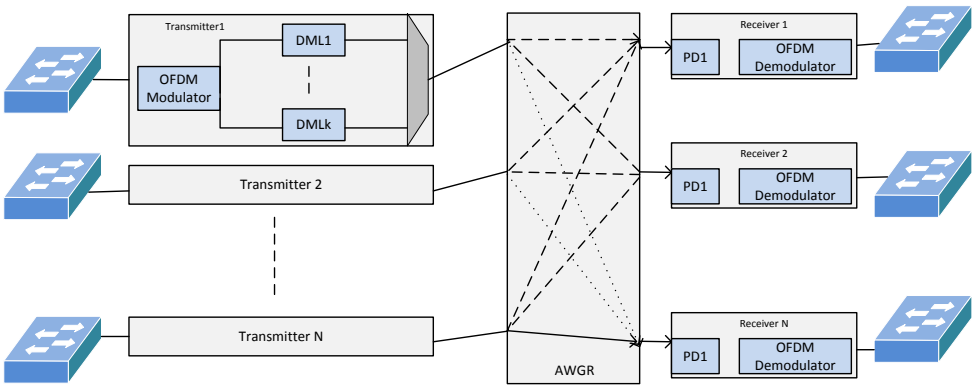


Figure 2.20. OFDM-based Architecture.

Parallel signal detection (PSD) technology [94] is used at the receiver in which photo-detector (PD) can concurrently distinguish many O-OFDM signals from multiple sources in different wavelengths, provided that there is no contention in the OFDM sub-carriers and the wavelengths. The centralized OFDM sub-carrier allocation scheme is proposed to avoid sub-carrier contention in any WDM channel at the receiver side. The centralized scheduler controls switching and is done by modulating the signal to new sub-carriers and turning on and off the lasers to generate new OFDM signal. The prototype is demonstrated in research paper [95].

The major advantage of this scheme is fast switching which results in low packet latency. The data rates can also be adjusted dynamically by using tunable lasers. The major drawbacks of this scheme are low scalability, less power efficient and high CAPEX cost. As it is based on, AWGR, so scalability is limited by the number of ports

of AWGR. The OFDM based transmitters are power hungry devices which increase power consumption of ToR switches.

In the next section, another types of architectures that are based on wavelength selective switches are discussed.

2.7 Architectures based on WSSs

2.7.1 Mordia

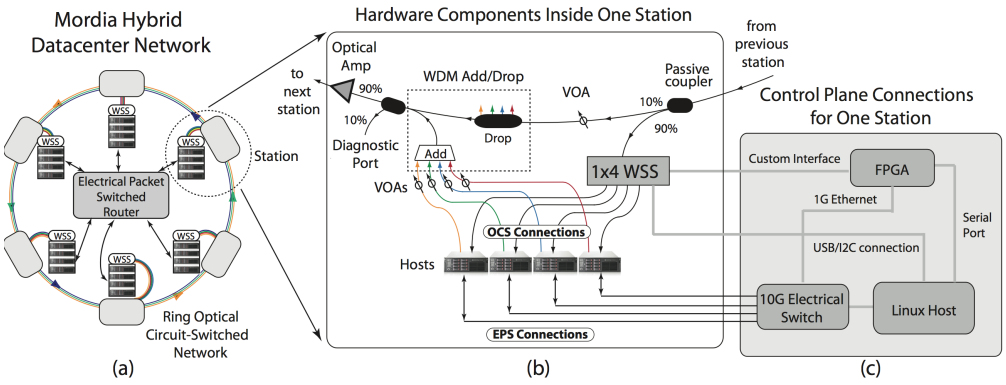


Figure 2.21. Mordia Architecture [6].

Mordia architecture was presented by [6, 31]. Its design is based on ring architecture and is shown in Figure 2.21(a). Each host has two interfaces, one is connected to traditional electronic packet switched network and other one is connected to OCS switch which is based on $1 \times k$ WSSs. It is a unidirectional ring of N individual wavelengths carried in a single fibre. Each host is assigned its own wavelength. Wavelengths are added or dropped from the OCS at each station. At each station, k wavelengths are added to the ring from each host at that station as shown in Figure 2.21(b). The input to each of WSS contains N wavelengths. The WSS selects k from N wavelengths and routes one each to the k hosts at that station.

The control plane of Mordia consists of a Linux host, FPGA board, and a 10G Ethernet switch as shown in Figure 2.21(c). Mordia uses time division multiple access (TDMA) by employing virtual output queues. TDMA divides the time into fixed-length

periods for data transmission and a time for circuit reconfiguration. The FPGA synchronizes the hosts and the WSSs by broadcasting a synchronization packet to all hosts over the EPS.

Scalability is the major limitation of Mordia which is limited by the size of WSSs and the number of wavelengths. As Mordia is based on ring architecture, so if one station or link goes down, the entire network gets affected. In the next section, WaveCube architecture [7] is presented that is also based on WSSs but it is scalable due to the usage of topology.

2.7.2 WaveCube

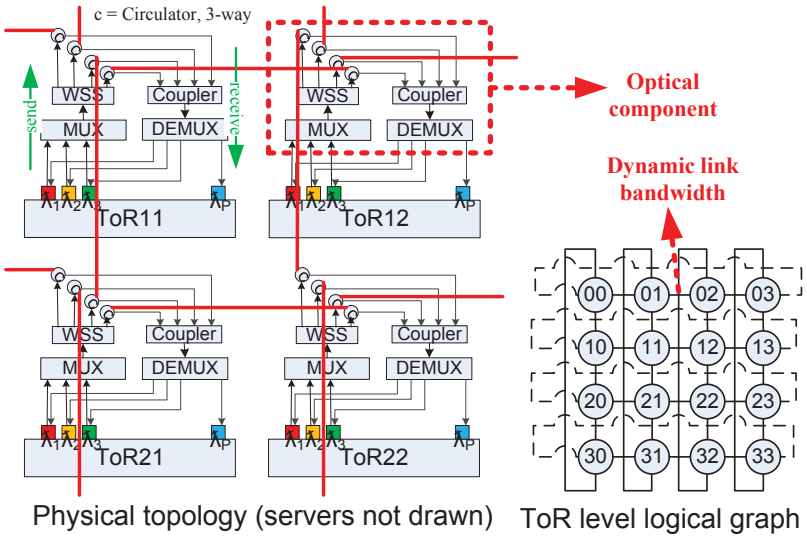


Figure 2.22. WaveCube Architecture [7].

WaveCube is an advanced version of OSA that does not employ MEMS switches [7]. WSSs are directly connected with each other without using MEMS. Its architecture is shown in Figure 2.22. It uses the same concept of multi-hopping and dynamic link bandwidth which was proposed in OSA. For dynamic link bandwidth, the researchers proposed a polynomial-time optimal algorithm to wavelength assignment. Every ToR switch is connected to k other ToR switches by a single hop while multi-hopping is used to achieve communication with other racks. The centralized control plane is used to achieve routing and wavelength assignments. It maintains the connectivity, utilization

and wavelength distribution information for each ToR link. The connectivity is used for detecting failures, utilization for link bandwidth optimization, and wavelength distribution for wavelength adjustment optimization, respectively.

Although the WaveCube is based on WSSs that are not scalable but overall architecture is scalable due to topology and employment of multi-hopping technique. Unlike Mordia, it is fault-tolerant since there is no single point of failure and multiple node-disjoint paths exist between any pair of ToRs. This architecture might be suitable for low diversity traffic but the performance of high diversity traffic workload might not be suitable to run on this architecture due to involvement of multiple hops which not only increase latency but also the power consumption. Furthermore, WaveCube performs well for high stability traffic but if the traffic is highly dynamic, WaveCube's performance would be unpredictable [7].

In the next section, another optical interconnect for DCNs which is also based on WSSs is presented. It is also scalable and is based on incremental and modular design. It is described below.

2.7.3 Optical Pyramid Data center Network Architecture

Optical pyramid data center network (OPMDC) [8] consists of three types of WSS-based optical switching nodes in three tiers as shown in Figure 2.23. Tier-1 comprises a group of pods. Each pod consists of B reconfigurable optical add/drop multiplexer (ROADM) in which each ROADM is connected to a ToR switch. Wavelength optical cross connects (WXC) nodes provide high-bandwidth optical switching in the second and third tiers. OPMDC is recursively built based on a pyramid construct that contains a polygonal base with an odd number (B) of nodes that are mesh connected via ribbon fibre cables. A macro-pod consists of B tier-2 WXC nodes. Each tier-2 WXC node is down connected to a pod, and up connected to the apex of its pyramid.

The traffic within a pod is routed horizontally in tier 1 while the traffic to other pods is routed first vertically in tier 2, then horizontally in that pod. Similarly if the destination host cannot be reached through the tier 2 then the traffic is routed through

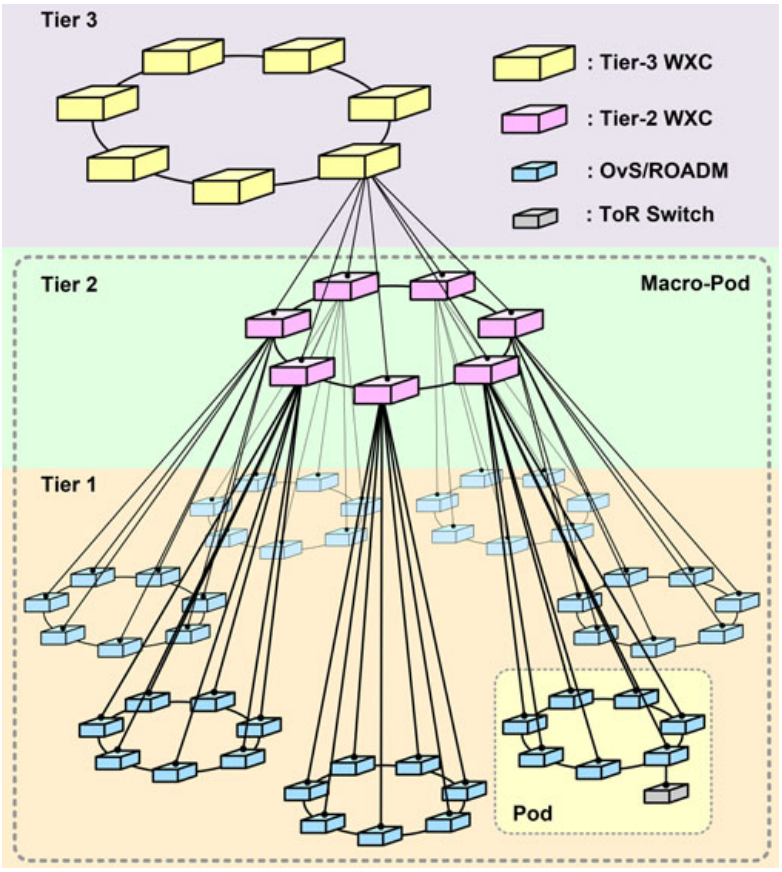


Figure 2.23. OPMDC Architecture [8].

the tier 3. OPMDC supports circuit switching for elephant flows and packet switching for mice flows.

OPMDC architecture is scalable due to its hierarchical design. However, implementation complexity and scalability of the control plane are the major concerns in this architecture. In the next section, hybrid architectures which are based on fast and slow optical switches are presented.

2.8 Architectures based on Fast and Slow Optical Switches

2.8.1 LIGHTNESS

The LIGHTNESS is a flattened architecture that employs OCS and OPS technologies [51, 52, 96–99] . Its architecture is shown in Figure 2.24. Servers are interfaced to ToR switches, which performs traffic aggregation and application-aware classification

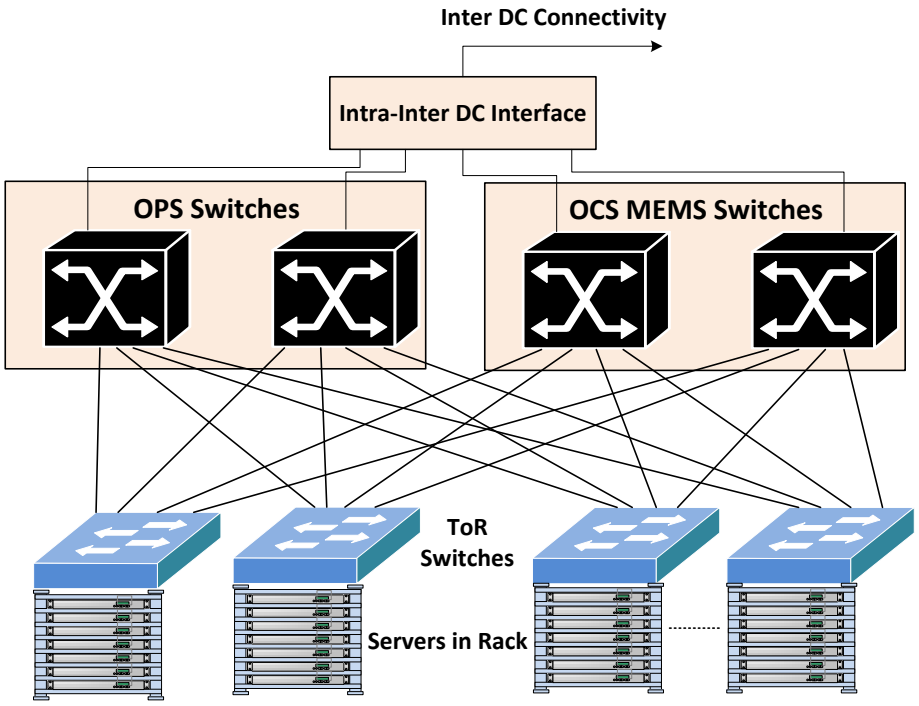


Figure 2.24. Lightness Architecture.

to either short- or long-lived traffic. OCS is used for long-lived connections and OPS is used for short-duration flows. Each ToR switch has interfaces to both OCS and OPS switches. OPS and OCS switches are also connected to a dedicated intra-to-inter data centre interface responsible to connect the DCN to metro or core networks for inter-data centre connectivity when required.

A unified SDN-based control plane architecture is proposed. The SDN control plane is responsible for the configuration of ToR switches and OCS/OPS switches. It also provides on-demand, flexible and programmable connectivity services for heterogeneous network functions and protocols.

The LIGHTNESS DCN can provide high capacity, low latency, high scalability, high programmability and flexible SDN control. However, the proposed SDN control plane comprises many functions and its implementation in the real world scenario is an open issue which will require complex hardware/software design and protocols. In the next section, hybrid optical switching scheme for DCNs [100, 101] that utilizes OCS, OPS and OBS is discussed. It is described below.

2.8.2 Hybrid Optical Switching

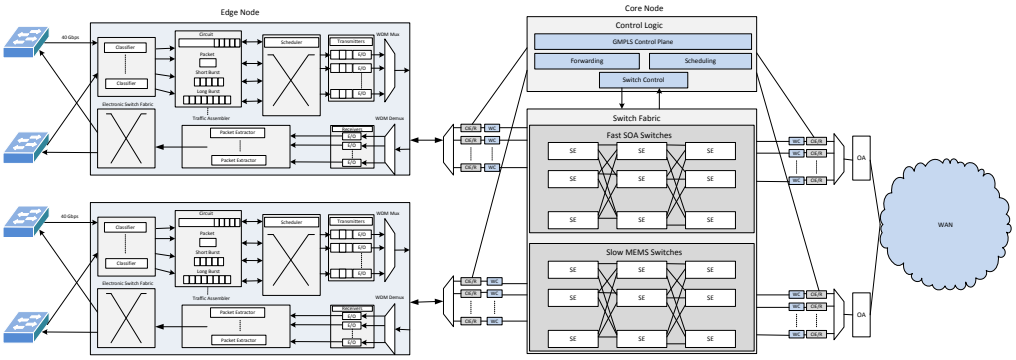


Figure 2.25. HOS Architecture.

Hybrid optical switching (HOS) scheme based on fast and slow optical switches has been presented by [100, 101]. Its architecture is shown in Figure 2.25. It consists of edge nodes and core nodes. Edge nodes are connected with ToR switches on one end, and on the other end, these are interfaced with core node using 40 Gbps links. Core nodes have control logic and switching fabric. The control logic consists of traditional GMPLS control plane and an HOS control plane. The HOS control plane performs forwarding, scheduling and switch configuration functions while GMPLS control plane performs routing, signalling and link management functions.

Switching fabric of HOS core node consists of fast and slow optical switches which are arranged into 3 stage clos-architecture. Slow switches are traditional MEMS switches while fast switches are based on SOAs arranged in Spank architecture [28]. Edge node classifies incoming traffic into four types i.e. circuit, long bursts, short burst and packets. Circuit and long bursts are routed through slow switches while short bursts and packets are routed through fast switches. TDM optical circuit switching is used for circuits establishment while optical burst switching is employed for long/short burst traffic. Packets are routed either to empty timeslots in already established TDM circuits or through the fast switches using packet switching. Since it uses burst and packet switching, burst and packet losses occur in case of contention in the network.

The burst and packet losses are the major limitations in this architecture which not only increase end to end packet latency due to retransmission but also decrease

overall throughput which is not feasible in the DCNs. Other drawback is CAPEX cost of the interconnect. Wavelength converters are used with every input/output port which makes overall architecture expensive solution.

From section 2.5 to 2.8, various optical interconnects, classified according to the type of optical switches, have been presented. A brief overview of each architecture along with its advantages and limitations has been provided. In the next section, a comparative analysis of these interconnects is presented.

2.9 Comparative Analysis

Table 2.1. Comparison at a Glance

Architecture	Year	Switching Technique	Capacity Limitation	Scalability	Cost	Power Efficiency	Implementation Complexity	Prototype
Architectures based on MEMS								
Helios	2010	OCS + EPS	Transc.	Low	Low	Low	Low	●
HyPaC	2010	OCS + EPS	Transc.	Low	Low	Low	Low	●
OSA	2014	OCS + EPS	Transc.	Medium	Low	High	Medium	●
Reconfigurable	2012	OCS + EPS	Transc.	High	Low	High	Medium	●
HydRA	2015	OCS + EPS	Transc.	High	Low	High	Medium	●
Architectures based on SOAs								
OSMOSIS	2004	OPS	SOA	Low	High	Low	Medium	●
Data Vortex	2008	OPS	SOA	High	High	Medium	Medium	●
SW	2011	OCS	SOA	Low	High	Low	High	
STIA	2011	OCS	SOA	Low	High	Low	High	
Bidirectional	2011	OPS	SOA	High	High	Medium	High	
Architectures based on AWGRs								
DOS & LIONS	2010 & 2013	OPS	TWC	Low	High	High	High	●
TONAK-LION	2013	OPS	TWC	High	High	High	High	
Petabit	2010	OPS	TWC	High	High	Medium	High	
IRIS	2010	OPS	TWC	High	High	Medium	High	●
OFDM-based	2013	OCS	TL	Low	High	Medium	High	●
Architectures based on WSSs								
Mordia	2013	OCS	WSS	Low	Medium	Medium	High	●
WaveCube	2015	OCS	WSS	High	Medium	Medium	High	●
OPMDC	2015	OCS	WSS	High	Medium	Medium	High	●
Architectures based on Fast and Slow Optical Switches								
LIGHTNESS	2013	OCS + OPS	Transc. + SOA	High	Medium	Medium	High	●
HOS	2014	OCS + OPS + OBS	Transc. + SOA	High	Medium	Medium	High	

This section briefly describes comparative analysis of proposed optical interconnects. Table 2.1 shows comparison of the proposed interconnects at a glance. It is shown in Table 2.1 that majority of the interconnects have been proposed the last 5 years, this shows that optical interconnects for DCNs have gained significant attention recently. The proposed interconnect schemes are still at an early stage of their development and there is lot of potential in this area of research to investigate new optical

interconnects technologies.

In table 2.1, it can be seen that the schemes which are based on hybrid electrical and MEMS switches, and schemes based on MEMS switches with multi-hoping technique rely on both EPS and OCS. OCS is used with MEMS switches and EPS is used with electrical switches or edge switches for multi-hop paths. OCS with MEMS switches is feasible because these switches are configured in tens of milliseconds and long lived traffic flows can tolerate delay in advance due to circuit establishment. The circuit establishment delay usually equals to the configuration delay of the MEMS switches because processing and propagation delays are negligible as compared to configuration delay of MEMS switches in DCNs.

Most of the schemes based on AWGR and SOAs use OPS. These schemes use TWCs/TLs or array of fixed lasers which are fast enough to process a packet within a few nanoseconds range. Limited buffering (FDLs or electrical buffers) is provided to process packet for routing decision or in case of network congestion. Packet losses can occur in these schemes due to network congestion or buffer overflow. Packet-based optical switching is feasible in situations where the duration of a flow between two nodes is very small. SW, STIA and OFDM-based techniques are the exceptions which use OCS. In SW and STIA, circuit is established for every packet i.e. before sending packet, the switching fabric is configured by the scheduler while in OFDM-based, packets are aggregated into a frame similar to OBS and circuit is established for every frame. In these schemes, circuit establishment delay relies heavily on processing and propagation delay because switching delay is negligible which is in nano seconds range.

Interconnection schemes which are based on WSSs use OCS. OPS is not feasible to use with WSSs because switching time of WSSs is in the range of microseconds to millisecond which is not fast enough to process a packet. Hybrid architectures rely on hybrid techniques. For example, LIGHTNESS uses OCS and OPS while HOS uses all three optical switching techniques.

The capacity limitation is an important feature when considering data centre upgradation. Interconnects based on MEMS can be easily upgrade to higher capacities

i.e. 40 Gbps, 100 Gbps or even higher rates because MEMS are independent of data-rate. The schemes based on AWGR depend on the maximum supported data-rate of TWCs/TLs. Data-rate is usually adjustable from low to high in TWCs/TLs. The High data-rates require high power consumption as compared to low data-rate. Similarly, the schemes based on SOAs and WSSs also depend on the data-rate of SOAs and WSSs. The hybrid schemes which are based on MEMS and SOAs, transceivers connected with MEMS switches are upgradable while transceivers connected with SOAs are data-rate dependent of SOAs.

Scalability is another important feature for DCNs. In MEMS based interconnects, Helios, HyPaC and OSA are not scalable while Reconfigurable and HydRA are scalable. This is due to the use of multiple MEMS switches in the Reconfigurable and HydRA architectures. Similarly in other designs, some architectures are scalable and some are not. It can be inferred from Table 2.1 that the scalability depends more on the usage of topology than on the usage of the types of optical switches.

Cost and power consumption are the next factors to consider when designing optical interconnects for DCN. Here the cost refers to the CAPEX cost while the operation expenditure (OPEX) cost relates with power efficiency. It can be seen from Table 2.1 that only MEMS based interconnects are low cost while interconnects based on AWGRs and SOAs are expensive. The cost of WSSs and hybrid designs are in medium range. The best design in terms of these two parameters is the one who has low CAPEX cost and is power efficient as well.

Implementation complexity and prototype design also relates with each other. In MEMS based interconnects, all schemes are easy to implement due to commercially available optical components. In all other schemes, the implementation complexity is high because optical components are not easily available or they are very expensive. However, some designs have shown prototype development on small scales as shown in Table 2.1.

2.10 Conclusion

This chapter gives a brief overview of various types of optical switches. Afterwards, a comprehensive survey of different types of optical interconnects for DCNs which have been proposed in recent years is given. The optical interconnects are classified into four types and comparative analysis of these interconnects is presented in the end.

After reviewing various optical interconnects, it can be said that every architecture has some pros and cons. The more feasible architecture would be the one which is scalable, power efficient, cost effective and can also provide low latency and high throughput.

In this thesis, three novel optical interconnection schemes for DCNs are proposed. The first two schemes are based on a hybrid design that utilizes fast and slow optical switches while the third scheme is based on only fast optical switches. Instead of using OCS/OPS/EPS, this research considers OBS in the proposed interconnects. A two-way reservation protocol of OBS is used that ensures zero burst loss. The proposed interconnects are highly scalable, power efficient and can provide low latency and high throughput. More detailed description of these interconnects is available in upcoming chapters of this thesis.

CHAPTER 3

HYBRID OPTICAL SWITCH ARCHITECTURE: HOSA

3.1 Introduction

In Chapter 2, various optical interconnects for DCNs that have been proposed in recent years are presented. Every design has some advantages and limitations. The ideal architecture should have some features like high scalability, power efficiency, low latency, high throughput and cost effective. By targeting these features, three architectures of optical interconnects for DCNs are proposed in this thesis.

This chapter introduces a novel optical interconnect for data centre networks which is based on fast and slow optical switches. The proposed technique leverages strengths of both types of optical switches. The strengths of one type of optical switch compensate for the weaknesses of the other type. The hybrid architecture features MEMS Optical Cross Connects (OXCs) for low cost. The low latency is achieved using fast optical switches.

The core of the innovation is to route traffic through a fast optical switch during the reconfiguration of a slow optical switch. The traffic is moved back on the slow MEMS switch once it is reconfigured. In this way, by using fast optical switch during reconfiguration time of slow optical switch should give overall switching time of the order of fast optical switch. A single-stage core topology with multiple optical switches allows the proposed design to both incrementally scaled up and scaled out without requiring major re-cabling.

Instead of using OCS or OPS, OBS is considered in this research. The OCS paradigm has been used in the backbone optical core network for many years. The OBS was also proposed for the backbone optical core network but it has not replaced OCS due to its limitation of high burst loss in this application. The OBS with a two-way reservation protocol ensures zero burst loss. The two-way reservation is not suitable for long-haul backbone optical networks due to the high RTT of the control packet but for the proposed optical interconnect for the DCN, this RTT is not high for several reasons presented in Chapter 1.

In proposed OBS mechanisms for DCNs, packets are aggregated for a certain period of time. A control packet is created to request the allocation of resources needed to transmit the data from the controller by using a two-way reservation process. The controller assigns resources and sends the control packet back to the originating node as an acknowledgement. Two bursts are formed i.e. one each for fast and slow optical switch paths if both paths are assigned by the controller. Otherwise, only one burst is formed either for fast or slow switch path assigned by the controller. The created bursts/burst are/is then transmitted on the pre-established paths/path configured by the controller.

The scalability analysis of the proposed architecture by considering different capacities of servers in a rack with different ratios of fast and slow optical switches is presented in this chapter. This chapter also determines a trade-off between cost and power consumption of the proposed design by comparing its cost and power consumption to those of well known interconnects. Furthermore, the performance evaluation of the system is done using network-level simulation by considering various traffic

workloads, and by using different capacities of slow and fast optical switches.

3.2 Hybrid Optical Switch Architecture: HOSA

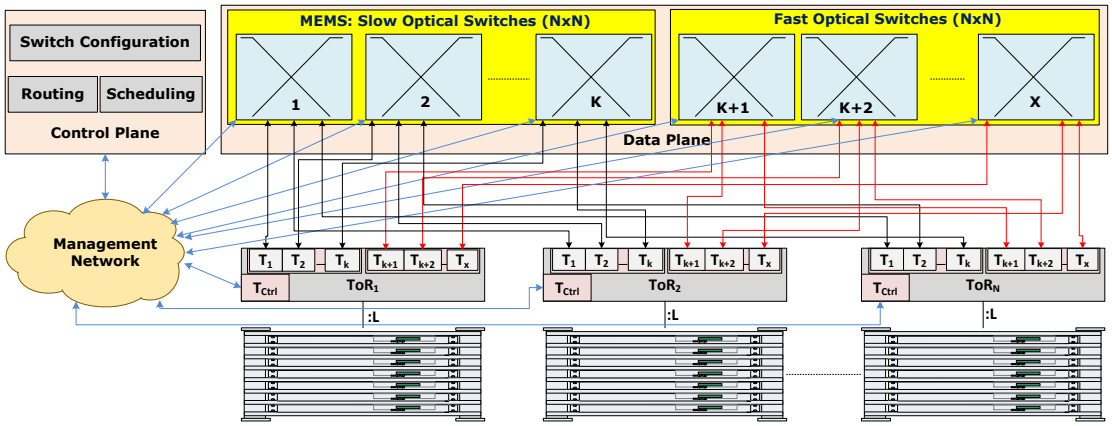


Figure 3.1. Proposed Architecture: HOSA

The proposed hybrid optical switch architecture (HOSA) for DCNs is shown in Figure 3.1. It is based on a two layer topology comprising electrical ToR switches at the edge and an array of optical switches at the core. The optical switches include both slow and fast optical switches. Servers in a rack are connected to ToR switches using bidirectional fibre links. Each ToR switch has X optical transceivers, in which K transceivers are linked to the slow optical switches and $X - K$ transceivers are interfaced to the fast optical switches, where $1 < K < X$. If we consider N as the total number of ToR switches in the network, then $(N \times N)$ is the minimum configuration for both fast and slow optical switches so that at least one port from all ToR switches connects to every $(N \times N)$ optical switch.

HOSA features separate data and control planes. The control plane is realized by using a centralized controller. Routing, scheduling and switch configuration are the main tasks of the controller. It handles connection requests from all ToR switches, finds routes to the destination ToR switch through optical switches, assigns timeslots to the connection requests by selecting a suitable link to the destination ToR switch, and configures optical switches with respect to the timeslots allocated. In order to realize these functions, the controller maintains a record of the global connectivity state of

the optical switches. The data plane is realized by using optical switches, performing data forwarding on pre-established light-paths configured by the controller. Each ToR switch has a dedicated optical transceiver which is connected to the controller through a management network.

3.2.1 Assumptions

This thesis assumes $(N \times N)$ size of both fast and slow optical switches in N rack DCN. This $(N \times N)$ size of optical switches for a very large value of N exist in the literature. For example, slow optical MEMS switches with more than a thousand ports have been proposed in research while in commercial configurations they exist only in 320 and 384 ports [26, 56, 102]. In the case of fast optical switch, 1024 ports optical switch using SOA-based switching fabric exist in the literature [29] while it is not available commercially. Large footprint and high cost are the major concerns in SOA-based switches. Photonic integration is only a viable solution for SOA-based switches to make them available in a large switching fabric. However, this is a challenging task and currently only 16×16 SOA based switches exist in the literature [67,68]. Similarly, AWGR switch with 512 ports also exist in the literature [60] but commercially it is available only in 32×32 standard configuration [61, 62]. Furthermore, AWGs with 128 channels are also available commercially [63], so a router module of AWGR with 128×128 configurations can be made by arranging them in a cyclic way as described in Chapter 2.

A lot of research is going on PICs and it is hoped that fast optical switches with a large scale switch configuration will be available at some time in the future. There is a possibility of integrated photonics leading to fast switches being in mass production and their integration to complementary metal oxide semiconductor (CMOS) circuits.

Due to availability of fast optical switches in small sizes, this thesis also discusses alternate approaches that can be considered using a small size of fast optical switches in combination with a large size of slow optical switches in the last chapter.

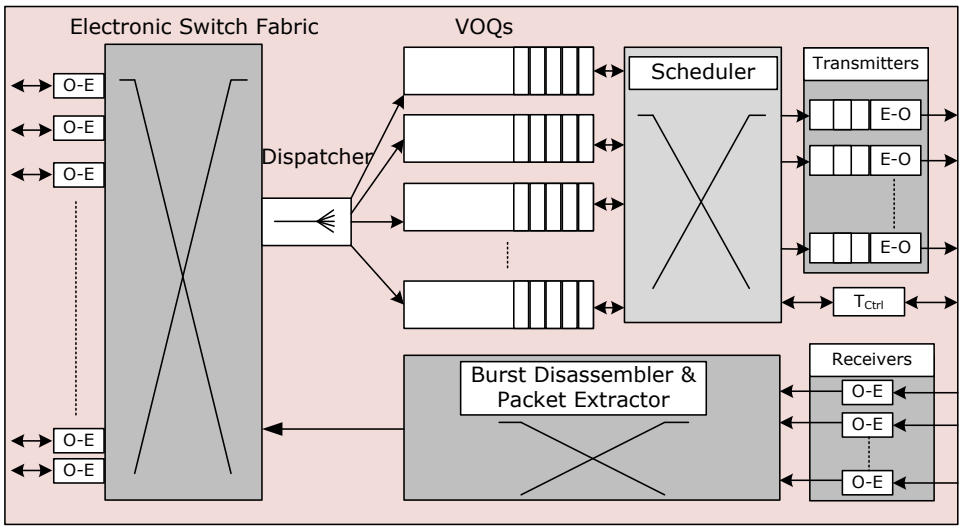


Figure 3.2. ToR Switch Design

3.2.2 ToR Switch Design

The ToR switch design is shown in Figure 3.2. It includes an electronic switch fabric which is connected to the servers in the rack to perform intra-rack (within rack) switching in the electrical domain. To perform inter-rack (between racks) switching, it employs $(N-1)$ Virtual Output Queues (VOQs) where N is the number of ToR switches in the network. The state of the art ToR switches support hundreds of VOQs. For example, the Cisco Nexus 5500 supports upto 384 VOQs, the Cisco 5548P supports upto 18432 VOQs and the Cisco 5596 supports upto 37728 VOQs [103, 104]. There is a VOQ for each destination ToR switch in the DCN. Packets destined to the same ToR are aggregated into the same VOQ. The VOQ not only aggregates traffic to the identical destination ToR switch but it also avoids Head Of Line blocking (HOL). Each VOQ is configured for a destination network address. Each ToR switch maintains a VOQ table where entries contain the destination rack network address and the VOQ number. The dispatcher module matches the destination network address of the packet with the entry in this table and forwards the packet on the required VOQ.

3.2.3 Dynamic Allocation of VOQs

For a very large scale DCN, the number of VOQs provided by the ToR switch can be less than the total number of racks in the DCN e.g. in the case of the Cisco 5500 switch, it supports only 384 VOQs. In this case we can use a subset of the total racks with which a given ToR communicates over a specified period of time. For example, in a thousand rack network, each rack may communicate with only a few other racks over a given period of time. So each rack would not require 999 VOQs. In this case we can dynamically allocate VOQs for the destination rack which the ToR is sending traffic to.

For dynamic allocation, we need another field, i.e. timestamps, in the VOQ table. In the first stage, the VOQ table contains only a list of VOQ entries without any corresponding destination network addresses. When a packet arrives at the dispatcher module, it looks up the destination network address in the list of VOQs but it does not find any match. Then it takes the first empty entry from the list of VOQs and assigns the destination network address and updates this field with the current timestamps and forwards the packet to this VOQ. When another packet arrives requesting the same destination rack, the dispatcher finds a match for this network address in the table, it updates its entry with the new timestamps and forwards the packet to the same VOQ. There is also a daemon process in the ToR switch that checks the VOQ list after a particular time interval. If the VOQ entry in the list has not been updated for that particular time, the destination network address entry from the list is deleted on the assumption that there is no more traffic for the destination rack. In this way, this VOQ can be assigned to another destination rack, after a timeout.

3.2.4 Control Packet Format for HOSA

The format of the control packet is shown in Figure 3.3. The control packet is 456 bits long and contains two main fields, routing and reservation. The routing field contains source and destination IP addresses of the ToR switches. These are the IP addresses of network interface cards (NICs) reserved for the control plane in ToR switches. The

value of 128 is considered for IPv6 address, however if IPv4 scheme is used, then this length can be reduced to 32 bits.

The reservation field is 176 bits long, and is divided into 7 sub-fields. The "Burst Length" field is filled by the ToR switch to inform the controller about the length of the burst it wants to send. The burst length is expressed in bytes. The rest of the 6 sub-fields are filled by the controller during processing of the control packet. The controller calculates time slots on the basis of the burst length requested by the ToR switch and assigns start time under "Fast/Slow Channel Starting Time" fields for both fast and slow switches. It also assigns the length of the burst under "Fast/Slow Path Burst Size" fields for both fast and slow switch paths. After assigning start time and burst size in respective fields, the controller assigns "Fast/Slow Path Channel No" that represent selected ports of the ToR switch in fast and slow switch paths.

If the requested burst length is divided between fast and slow switch paths then the controller fills all the sub-fields in the reservation field. If the requested length is assigned only to fast or to slow switch path, then only relevant fields are filled and rest of the fields are ignored. The CRC field is reserved for Cyclic Redundancy Check and a couple of optional fields are reserved for flags.

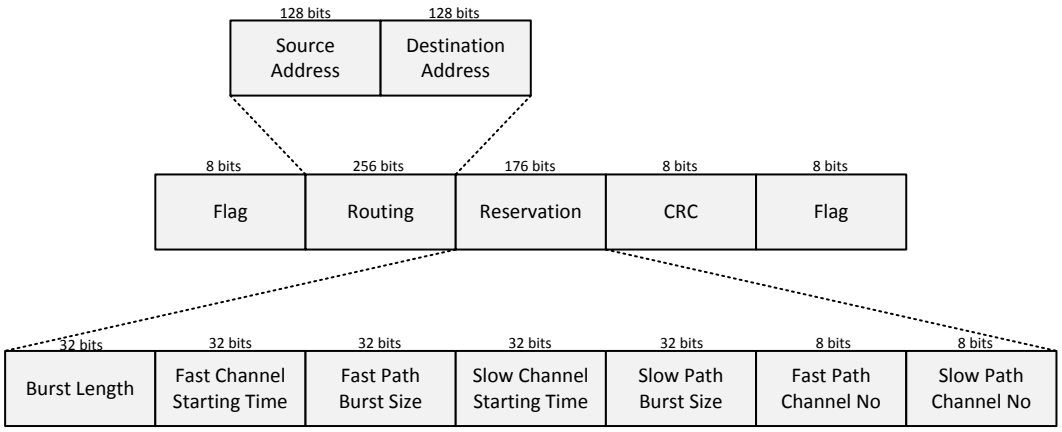


Figure 3.3. Control Packet Format for HOSA

Algorithm 1 Control Packet Generation at ToR Switches for HOSA

Require: $timeout \leftarrow timeoutParameter$
{**timeoutParameter is required during ToR switches configuration.**}
1: $timeoutevent = NULL$
2: $burst\_length = 0$
{**Above two lines initialize** $timeoutevent$ **and** $burst\_length$ **parameters.**}
{**Following condition is used to check a timeout event or a packet arrival.**}
3: **if** $timeoutevent$ **then**
4: $control\_packet = generateControlPacket()$
{**Above line generates a control packet.**}
5: $control\_packet.setBurstLength(burst\_length)$
6: $send(control\_packet, T_{ctrl})$
{**timeoutevent is for timer check. This block is executed when timeoutevent occurs i.e. timer expired. The control packet is generated and is sent to the management network.**}
7: **else**
8: $packet \leftarrow packet\ arrives$
9: **if** $VOQ \rightarrow empty()$ **then**
10: $firstpk\_time = current\_time$
11: $schedule(firstpk\_time + timeout, timeoutevent)$
{**Schedule timeoutevent by adding timeout parameter in first packet arrival time.**}
12: **end if**
13: $burst\_length += packet.length$
14: $VOQ.insertPacket(packet)$
{**Add packet in virtual output queue.**}
15: **end if**

3.2.5 Burst Assembly/Disassembly

The timer based algorithm is considered in VOQs to aggregate traffic. Timer starts when the first packet comes at the VOQ and when it expires, ToR switch generates a control message and sends it to the controller for new reservation. The controller assigns new timeslots and sends the control packet back to the source ToR switch. It also generates a control packet for the destination ToR switch to enable bidirectional communication. After receiving the control packet, ToR switch creates and schedules bursts to the specified timeslots.

The timer-based algorithm for the generation of control packets is shown in Algorithm 1. The timer starts when a packet comes at the empty VOQ (lines 8-11 in Algorithm 1). If the VOQ is not empty when the packet arrives, it joins other packets

in the VOQ (lines 13-14 in Algorithm 1). Upon expiration of the timer, the control packet is created and is sent to the controller to request a new timeslot (lines 3-6 in Algorithm 1). At this stage, the control packet comprises the requested burst length in bytes and the IP addresses of source and destination ToR switches. The controller performs routing and scheduling algorithm as described in Algorithm 3 and sends it back to the ToR switch.

The control packet now also contains the ToR port numbers, start time and the burst lengths in slow/fast optical switch paths assigned by the controller. The scheduler in the ToR switch generates two bursts, one each for fast and slow path if both fast and slow switch timeslots have been allocated, otherwise it generates one burst either for slow or fast switch path allocated by the controller. The generated bursts are then forwarded to the queue of the allocated ports. This procedure is called burst generation and is shown in Algorithm 2. The relevant information from the control packet is extracted by the scheduler of the ToR switch (lines 1-9 in Algorithm 2). The burst in fast switch path is generated and transmitted if timeslot has been allocated by the controller in the fast switch path (lines 10-21 in Algorithm 2). Similarly, the burst in slow switch path is generated and transmitted if timeslot has been allocated by the controller in the slow switch path (lines 22-33 in Algorithm 2). The scheduler starts a new timer if the VOQ is not empty after burst transmission because new packets might have arrived during the RTT of the control packet (lines 38-41 in Algorithm 2).

Algorithm 2 Bursts Generation at ToR Switch for HOSA

Require: $timeout \leftarrow timeoutParameter$
{**timeoutParameter is required during ToR switches configuration.**}
1: $cp \leftarrow control\ packet\ arrives$
2: $fb \leftarrow generateBurst()$
3: $sb \leftarrow generateBurst()$
4: $fbsize \leftarrow cp \rightarrow getfbSize()$
5: $sbsize \leftarrow cp \rightarrow getsbSize()$
6: $fc \leftarrow cp \rightarrow getfcNo()$
7: $sc \leftarrow cp \rightarrow getscNo()$
8: $fst \leftarrow cp \rightarrow getFastStatTime()$
9: $sst \leftarrow cp \rightarrow getSlowStartTime()$
{**Above lines initialize different variables when a control packet arrives at the ToR switch after being processed by the controller.**}
10: **if** $fc \neq NULL$ **then**
11: **while** $VOQ \rightarrow hasPackets()$ **do**
12: **if** $fb \rightarrow length \leq fbsize$ **then**
13: $packet \leftarrow VOQ \rightarrow getPacket()$
14: $fb \rightarrow length += packet \rightarrow length$
15: $fb \rightarrow insertPacket(packet)$
16: **else**
17: $break$
18: **end if**
19: **end while**
{**Above block inserts packets from VOQ into the generated burst fb for fast path according to its size.**}
20: $schedule(fb, fc, fst)$
{**Above line schedule burst fb in fast path on allocated port fc at time fst.**}
21: **end if**
22: **if** $sc \neq NULL$ **then**
23: **while** $VOQ \rightarrow hasPackets()$ **do**
24: **if** $sb \rightarrow length \leq sbsize$ **then**
25: $packet \leftarrow VOQ \rightarrow getPacket()$
26: $sb \rightarrow length += packet \rightarrow length$
27: $sb \rightarrow insertPacket(packet)$
28: **else**
29: $break$
30: **end if**
31: **end while**
{**Above block inserts packets from VOQ into the generated burst sb for slow path according to its size.**}
32: $schedule(sb, sc, sst)$
{**Above line schedule burst sb in slow path on allocated port sc at time sst.**}
33: **end if**

34: **if** $VOQ \rightarrow empty()$ **then**
35: $burst\_length = 0$
36: $firstpk\_time = 0$
37: **else**
38: $pk \leftarrow VOQ \rightarrow get(0)$
39: $firstpk\_time \leftarrow pk.arrivaltime$
40: $schedule(firstpk\_time + timeout, timeoutevent)$
41: $burst\_length = VOQ \rightarrow getTotalPacketsLength()$
42: **end if**
{**Lines 34 to 42 are used to reset various variables after burst scheduling.**}

In order to realize bidirectional communication, the controller also generates a control packet and sends it to the destination ToR switch. The timer of the VOQ in the destination ToR might have not expired but as soon as it receives the control packet, it generates and transmits bursts/burst as described in Algorithm 2.

When the burst arrives at the ToR switch via receivers, the burst disassembler module as the name suggests, disassemble the burst, extracts packet from the burst and sends them to the electronic buffer in electronic switch fabric to forward these packets to different ports in the rack.

3.2.6 Routing and Scheduling

The controller keeps a record of the connections of all optical switches. It performs routing, scheduling and optical switch configuration. The routing and scheduling mechanism is described in Algorithm 3.

The control packet sent by the ToR switch arrives at the controller for timeslot request. The control packet at this stage contains information of requested Burst Length (BL), a source IP address and a destination IP address. The controller maintains a routing table which has information regarding physical connections of all optical switches with ToR switches and a horizon table which contains record of connectivity states of all channels. The proposed technique employs a horizon-based scheduling algorithm similar to that presented for optical burst switching networks [42]. The term horizon refers to the latest available time when the channel will be free. There are also some other scheduling techniques in OBS networks but this technique is considered in this

Algorithm 3 Routing and Scheduling Algorithm for HOSA

1: $cp \leftarrow controlpacket$
{**Above line shows that a control packet arrives at the controller and is assigned to** cp **object.** }
2: $srcID \leftarrow getSrcId(cp \rightarrow getSourceAdd)$
3: $destID \leftarrow getDestId((cp \rightarrow getDestAdd)$
{**Above two lines extract source and destination addresses from the control packet and finds their relevant IDs and assign them to two variables (**$srcID$ **and** $destID$**).**}
4: $T_{pre} \leftarrow previousReservationTime(srcID, destID)$
5: **if** $(T_{cur} - T_{pre}) < T_{dup}$ **then**
6: $delete(cp)$
{**Condition in line 5 is used to avoid duplicate timeslot allocation. If true then the control packet is deleted because ToR pair has been assigned a timeslot recently.** }
7: **else**
8: $src\_fc \leftarrow findH\_Ch\_FS(srcID)$
9: $dest\_fc \leftarrow findH\_Ch\_FS(destID)$
10: $src\_sc \leftarrow findH\_Ch\_SS(srcID)$
11: $dest\_sc \leftarrow findH\_Ch\_SS(destID)$
{**Above four lines perform routing operation and return channels of latest horizon using minimum value search function. The routing operation results in finding total 4 channels in which there are 2 channels for source (1 for slow (**src_sc**) and 1 for fast path (**src_fc**)) and 2 channels for destination ToR (1 for slow (**$dest\_sc$**) and 1 for fast path (**$dest\_sc$**)).** }
12: $src_{est} \leftarrow findH\_Ch\_SSest(srcID, destID)$
{**In above line,** $findH\_Ch\_SSest(srcID, destID)$ **function is used to find an already established slow switch path between ToR pair which is assigned to** src_{est} **variable.** }
13: $T_{fast} \leftarrow getMax(getH\_F(src\_fc), getH\_F(dest\_fc))$
14: $T_{slow} \leftarrow getMax(getH\_SS(src\_sc), getH\_SS(dest\_sc))$
{**In above two lines,** $getMax(value1, value2)$ **function is used to get the maximum value,** $getH\_F(channel)$ **function is used to get the horizon for fast path and** $getH\_SS(channel)$ **function is used to get horizon of slow switch paths. The maximum horizons for fast and slow switch paths are assigned to** T_{fast} **and** T_{slow} **respectively.**}
15: **if** $T_{fast} < T_{cur}$ **then**
16: $T_{fast} \leftarrow T_{cur}$
17: **end if**
18: **if** $T_{slow} < T_{cur}$ **then**
19: $T_{slow} \leftarrow T_{cur}$
20: **end if**
{T_{fast} **and** T_{slow} **are assigned a current time if they are less than the current time.** }

21: **if** $src_{est} \neq NULL$ **then**
22: $T_{est} \leftarrow getH\_SS(est)$
23: **if** $T_{est} < T_{cur}$ **then**
24: $T_{est} \leftarrow T_{cur}$
25: **end if**
26: $dest_{est} \leftarrow getDest\_ch(src_{est})$

{**If condition in line 21 is false then it shows that an already established slow switch path does not exist. If it is true then the algorithm finds the horizon of already established slow switch path using** $getH\_SS(channel)$ **function and assigns it to** T_{est}**. The** T_{est} **is also assigned a** T_{cur} **if it is less than the current time. The** $getDest\_ch(channel)$ **method is used to get channel in destination ToR.**}

27: **if** $T_{est} > (T_{slow} + T_{sws})$ **then**
28: $T_{est} \leftarrow NULL$
29: **end if**
30: **end if**

{**If condition in line 27 is true then** T_{est} **is assigned a** $NULL$ **value. It means already established connection will not be considered for scheduling.**}

31: **if** $(T_{est} \neq NULL)$ **then**
32: $T_{slow} \leftarrow T_{est}$
33: $scr\_sc \leftarrow src_{est}$
34: $dest\_sc \leftarrow dest_{est}$
35: $T_{sws} \leftarrow 0$
36: **end if**

{**If condition in line 31 is true then already established connection will be considered for scheduling.** }

37: $BL \leftarrow cp \rightarrow getBurstLength()$

{**Above line extracts burst length from the control packet and assigns it to** BL **variable.** }

38: $T_{RL} \leftarrow BL * 8/datarate$

{**Above line calculates burst length in time and assigns it to** T_{RL} **variable.**}

39: **if** $(T_{fast} \leq T_{slow})$ **then**
40: **if** $((T_{RL} + T_{fast} + T_{swf} + T_{oh} + T_{proc}) \leq (T_{slow} + T_{sws} + T_{oh} + T_{proc}))$ **then**
41: $cp \rightarrow setFastPathChannelNo(src\_fc \pmod{fast\_ch}))$
42: $cp \rightarrow setFastPathBurstSize(BL)$
43: $cp \rightarrow setFastChannelStartingTime(T_{fast} + T_{swf} + T_{oh} + T_{proc})$
44: $setupFastPath(src\_fc, dest\_fc, T_{fast}+T_{swf}+T_{oh}+T_{proc}, T_{RL}+T_{fast}+T_{swf}+T_{oh}+T_{proc}+T_{guard}, T_{cur})$

{**If condition in lines 39 and 40 are true then a fast switch path is assigned as shown above.** }

45: $cpdest \leftarrow cp \rightarrow dup()$
46: $cpdest \rightarrow setFastPathChannelNo(dest\_fc \pmod{fast\_ch}))$

{**Line 45 creates a duplicate control packet** $(cpdest)$ **and line 46 sets the port number in the fast path.**}

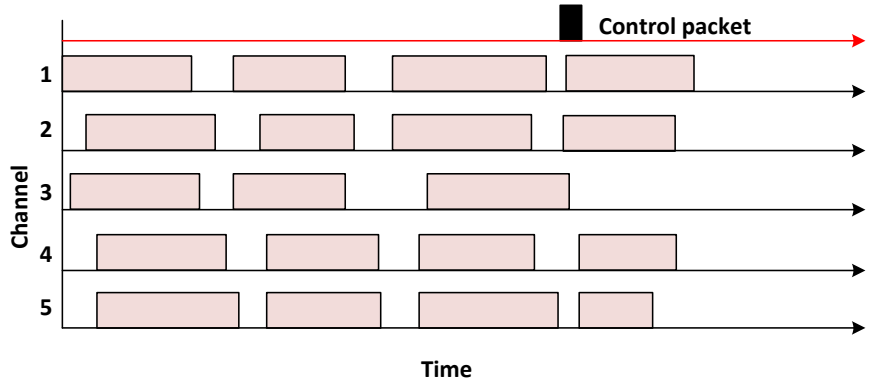
47: **else**
48: $sls \leftarrow (T_{fast} + T_{swf} + T_{oh} + T_{proc}) - (T_{slow} + T_{sws} + T_{oh} + T_{proc}) * datarate/8$
49: $cp \rightarrow setFastPathChannelNo(src\_fc \pmod{fast\_ch})$
50: $cp \rightarrow setFastPathBurstSize(BL - (sls))$
51: $cp \rightarrow setFastChannelStartingTime(T_{fast} + T_{swf} + T_{oh} + T_{proc})$
52: $setupFastPath(src\_fc, dest\_fc, T_{fast} + T_{swf} + T_{oh} + T_{proc}, ((BL - sls) * 8/datarate) + T_{fast} + T_{swf} + T_{oh} + T_{proc} + T_{guard}, T_{cur})$
53: $cp \rightarrow setSlowPathChannelNo(src\_sc \pmod{slow\_ch})$
54: $cp \rightarrow setSlowPathBurstSize(sls)$
55: $cp \rightarrow setSlowChannelStartingTime(T_{slow} + T_{sws} + T_{oh} + T_{proc})$
56: $setupSlowPath(src\_sc, dest\_sc, T_{slow} + T_{sws} + T_{oh} + T_{proc}, ((sls) * 8/datarate) + T_{slow} + T_{sws} + T_{oh} + T_{proc} + T_{guard}, T_{cur})$
{**Both fast and slow paths are assigned if condition in line 40 is false as describe above.**}
57: $cpdest \leftarrow cp \rightarrow dup()$
58: $cpdest \rightarrow setFastPathChannelNo(dest\_fc \pmod{fast\_ch})$
59: $cpdest \rightarrow setSlowPathChannelNo(dest\_sc \pmod{slow\_ch})$
60: **end if**
61: **else if** $(T_{slow} + T_{sws} + T_{oh} + T_{proc}) \leq (T_{fast} + T_{swf} + T_{oh} + T_{proc})$ **then**
62: $cp \rightarrow setSlowPathChannelNo(src\_sc \pmod{slow\_ch})$
63: $cp \rightarrow setSlowPathBurstSize(BL)$
64: $cp \rightarrow setSlowChannelStartingTime(T_{slow} + T_{sws} + T_{oh} + T_{proc})$
65: $setupSlowPath(src\_sc, dest\_sc, T_{slow} + T_{sws} + T_{oh} + T_{proc}, T_{RL} + T_{slow} + T_{sws} + T_{oh} + T_{proc} + T_{guard}, T_{cur})$
{**Slow path is assigned if condition in line 61 is true as describe above.**}
66: $cpdest \leftarrow cp \rightarrow dup()$
67: $cpdest \rightarrow setSlowPathChannelNo(dest\_sc \pmod{slow\_ch})$
68: **else**
69: $sls \leftarrow (T_{RL} + T_{fast} + T_{swf} + T_{oh} + T_{proc}) - (T_{slow} + T_{sws} + T_{oh} + T_{proc}) * datarate/8$
70: **if** $sls \leq 0$ **then**
71: $cp \rightarrow setFastPathChannelNo(src\_fc \pmod{fast\_ch})$
72: $cp \rightarrow setFastPathBurstSize(BL)$
73: $cp \rightarrow setFastChannelStartingTime(T_{fast} + T_{swf} + T_{oh} + T_{proc})$
74: $setupFastPath(src\_fc, dest\_fc, T_{fast} + T_{swf} + T_{oh} + T_{proc}, T_{RL} + T_{fast} + T_{swf} + T_{oh} + T_{proc} + T_{guard}, T_{cur})$
75: $cpdest \leftarrow cp \rightarrow dup()$
76: $cpdest \rightarrow setFastPathChannelNo(dest\_fc \pmod{fast\_ch})$
{**Fast path is assigned if condition in line 70 is true as describe above.**}

77: **else**
78: $cp \rightarrow setFastPathChannelNo(src\_fc \pmod{fast\_ch})$
79: $cp \rightarrow setFastPathBurstSize(BL-(sls))$
80: $cp \rightarrow setFastChannelStartingTime(T_{fast}+T_{swf}+T_{oh}+T_{proc})$
81: $setupFastPath(src\_fc, dest\_fc, T_{fast}+T_{swf}+T_{oh}+T_{proc}, (BL-(sls))*8/datarate+T_{fast}+T_{swf}+T_{oh}+T_{proc}+T_{guard}, T_{cur})$
82: $cp \rightarrow setSlowPathChannelNo(src\_sc \pmod{slow\_ch})$
83: $cp \rightarrow setSlowPathBurstSize(sls)$
84: $cp \rightarrow setSlowChannelStartingTime(T_{slow}+T_{sws}+T_{oh}+T_{proc})$
85: $setupSlowPath(src\_sc, dest\_sc, T_{slow}+T_{sws}+T_{oh}+T_{proc}, (sls*8/datarate)+T_{slow}+T_{sws}+T_{oh}+T_{proc}+T_{guard}, T_{cur})$
86: $cpdest \leftarrow cp \rightarrow dup()$
87: $cpdest \rightarrow setFastPathChannelNo(dest\_fc \pmod{fast\_ch})$
88: $cpdest \rightarrow setSlowPathChannelNo(dest\_sc \pmod{slow\_ch})$
{**Both slow and fast paths are assigned if condition in line 70 is false as describe above.**}
89: **end if**
90: **end if**
91: **end if**

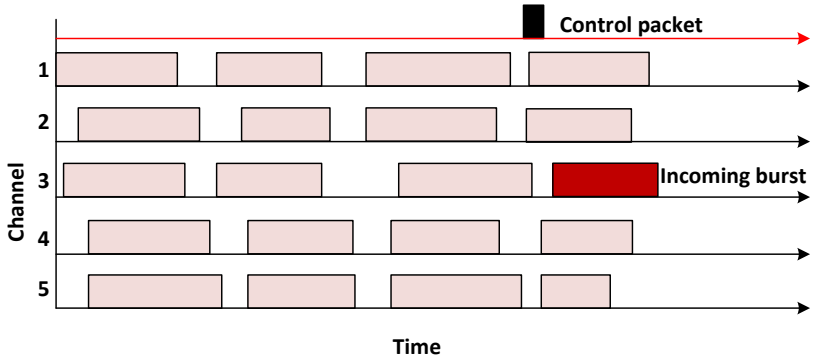
research due to its efficiency and its implementation simplicity. The goal is to use a technique that fulfils the objective of a fast optical control plane.

The concept of horizon scheduling is explained with the help of Figure 3.4. Let suppose there are 5 channels on which an incoming burst can be scheduled. Figure 3.4(a) shows the states of the channels when a control packet arrives at the controller. The horizon scheduling uses a minimum value method to find a latest available channel. Figure 3.4(b) shows the states of the channels after allocating a timeslot for the incoming burst using horizon scheduling.

In first step of the routing and scheduling algorithm, the controller extracts the source and the destination IP addresses of the ToR switches from the control packet and finds their relevant IDs (Lines 2-3 of Algorithm 3). The next step is to check whether the same ToR pair has been assigned a timeslot recently to avoid duplicate timeslot allocation. For example, ToR 1 and 2 send control packets to the controller at the same time or with very little time difference. The controller receives the first control packet and schedules it and sends it to both ToR 1 and 2. Meanwhile it also receives the second control packet. To avoid duplicate timeslot allocation, it deletes the control packet if the ToR pair has been assigned the timeslot recently. For this



(a)



(b)

Figure 3.4. Resource Allocation Mechanism using Horizon Scheduling, (a) Channel states before timeslot allocation and (b) Channel states after timeslot allocation.

purpose, three parameters are defined: T_{pre},T_{cur} and T_{dup}. The T_{pre} is the previous reservation time for the ToR pair, the T_{cur} is the current time and the T_{dup} is the time to avoid for duplicate allocation. The control packet is deleted if the difference of T_{cur} and T_{pre} is less than T_{dup}. The T_{dup} is a fixed parameter that should be greater than twice of the overhead time T_{oh}.

In the next step, the controller selects the optimal routes for slow and fast switch paths by using source and destination IDs of ToR switches in the control packet, then it finds an already established slow switch path between ToR pair as shown in lines 8-12 of Algorithm 3.

The controller assigns timeslots to the control packet after performing scheduling function. The basic principle of the scheduling function is that every request is as-

signed timeslots from slow path, fast path or from both slow and fast paths so that minimum end-to-end delay is ensured. First, the controller finds horizons of fast and slow paths. It is done by using minimum value search method from horizons of all the channels from source destination ToR switch pair. T_{fast} and T_{slow} (lines 13-14 of Algorithm 3) represent horizons of fast and slow channels respectively. The T_{fast} and T_{slow} are assigned a value of the current time T_{cur} if they are less than T_{cur} (lines 15-20 of Algorithm 3). T_{est} is the horizon of already established slow path. The T_{est} is assigned a value of the current time if it is less than T_{cur}. The T_{est} is assigned a $NULL$ value if $T_{est} > (T_{slow} + T_{sws})$ because latency in using established path will be higher as compared to establish a new slow path, where T_{sws} is the switching time of slow switch (lines 21-30 of Algorithm 3). In the next step, if $T_{est} \neq NULL$, then T_{slow} is replaced with T_{est} and their respective channels are also replaced and T_{sws} is assigned a zero value (lines 31-36 in Algorithm 3) because latency of establishing a new slow path will be higher as compared to using already established path. The requested timeslot length is calculated on the basis of burst length in the control packet. Requested timeslot length T_{RL} is given by:

$$T_{RL} = \frac{BL \times 8}{datarate} \tag{3.1}$$

where BL is the burst length specified by ToR switch in the control packet (lines 37-38 of Algorithm 3). The next step is to find a channel either from fast path or slow path or from both. The fast path is assigned if the conditions in line 40 and 70 of Algorithm 3 are satisfied, where T_{swf} is the switching time of fast switch, T_{oh} is the overhead time and T_{proc} is the processing time of the control packet at the controller. The controller fills ToR switch fast path channel number, fast path burst size and fast path timeslot in the control packet (lines 41-43,71-73) and sets up a fast path (line 44,74). The controller also creates a duplicate control packet and updates fast path channel number with respect to the destination ToR switch (lines 45-46,75-76 in Algorithm 3).

The slow path is assigned if the condition in line 61 of Algorithm 3 is satisfied. The controller fills ToR switch slow path port number, slow path burst size and slow path timeslot in the control packet (lines 62-64 in Algorithm 3) and sets up a slow

path (line 65 in Algorithm 3). The controller also creates a duplicate control packet and updates slow path port number with respect to the destination ToR switch (lines 66-67 in Algorithm 3).

Both slow and fast paths are assigned if the conditions in line 40 and 70 in Algorithm 3 are false. In this case, sls (line 48,69 in Algorithm 3) timeslot length is assigned to the slow path and $BL-sls$ (line 50,79 in Algorithm 3) timeslot is assigned to the fast path. Similarly, the controller fills the relevant fields in the control packet and sets up both fast and slow paths. It also generates a duplicate control packet and updates relevant information with respect to the destination ToR switch. In the end, the original control packet is sent to the source ToR switch and the duplicate control packet is sent to the destination ToR switch in order to realize bidirectional communication.

There is also an optional feature of the routing and scheduling algorithm which is called a speculation approach. So far the algorithm described above does not include this feature. In the routing and scheduling with the speculation approach, if the routing and scheduling algorithm fails to establish a new slow path then it performs additional check. A new slow path is assigned to a channel which has not been used for a particular time interval (T_{idle}). More specifically, the algorithm sets up a new slow path if $T_{cur} - T_{slow} \geq T_{idle}$. In this case T_{slow} will be the original horizon of the slow path i.e. it may be less than T_{cur}.

Figures 3.5,3.6,3.7,3.8,3.9 and 3.10 represent the timeslot allocation mechanism described above. There are two parts in each figure, (a) represents channel states before timeslot allocation i.e. when a control packet arrives at the controller and (b) represents channel states when the timeslot has been allocated for requested timeslot duration. Each figure comprises three channels for fast switch paths and three channels for slow switch paths.

Figure 3.5 shows a first scenario in which timeslots are allocated in both fast and slow switch paths. For example, a timeslot in a fast switch path is assigned (i.e. first channel in the fast switch paths) during the reconfiguration time of the slow switch T_{sws} and a timeslot in a slow switch path (i.e. 2nd channel in the slow switch paths)

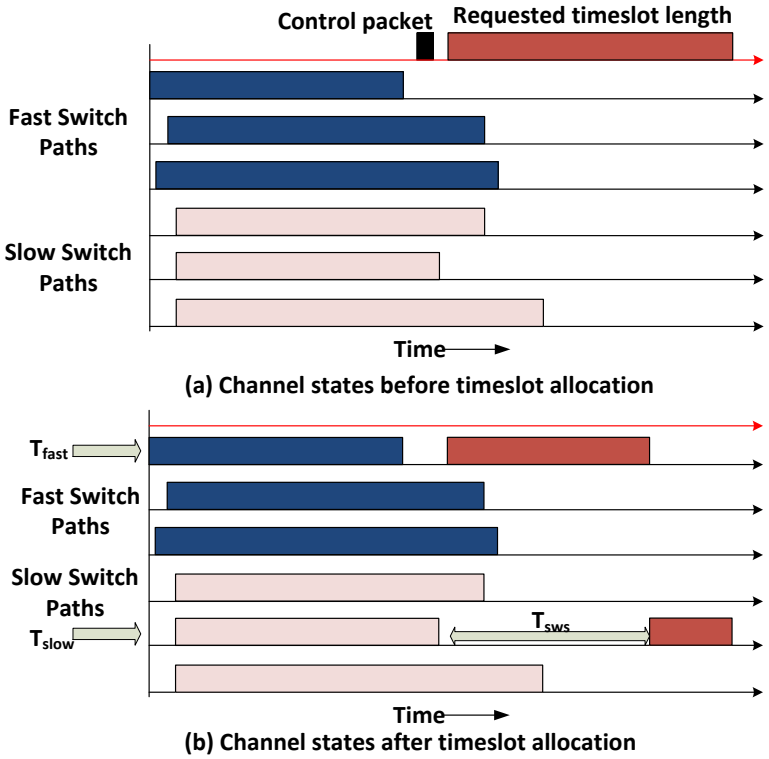


Figure 3.5. Timeslot Allocation for HOSA: Case 1

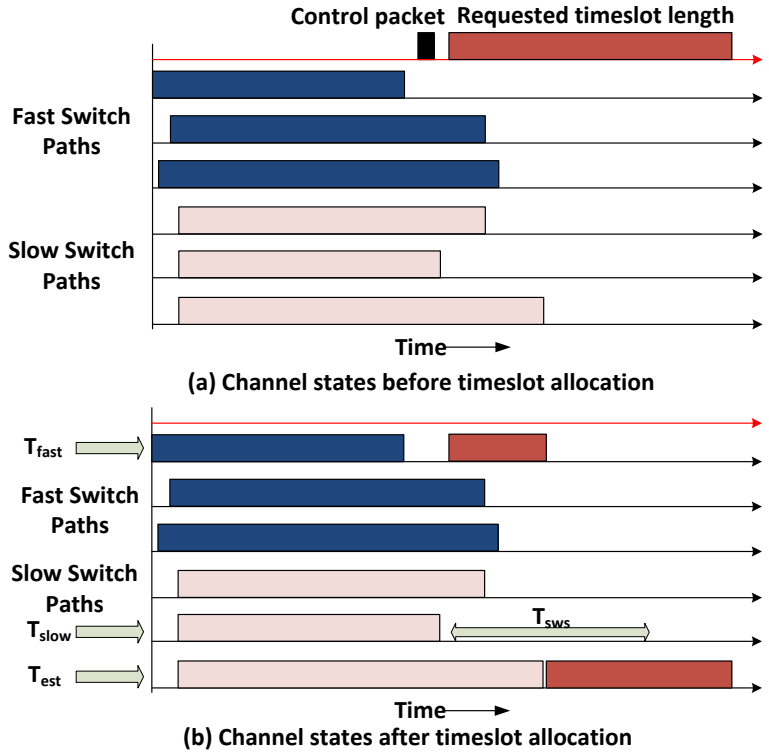


Figure 3.6. Timeslot Allocation for HOSA: Case 2

is assigned after the T_{sws}. In this way, the latency of T_{sws} is avoided by assigning a timeslot in the fast switch path during T_{sws}.

Figure 3.6 represents a second scenario when there is an already established slow switch path exists between ToR pairs. T_{est} is the horizon of an already established slow switch path. In this scenario, timeslots are allocated in fast and in already established slow paths i.e. first channel of the fast switch paths and 3rd channel of the slow switch paths. In this way, by utilizing an already established slow switch path, the bandwidth which will be wasted while T_{sws} is avoided.

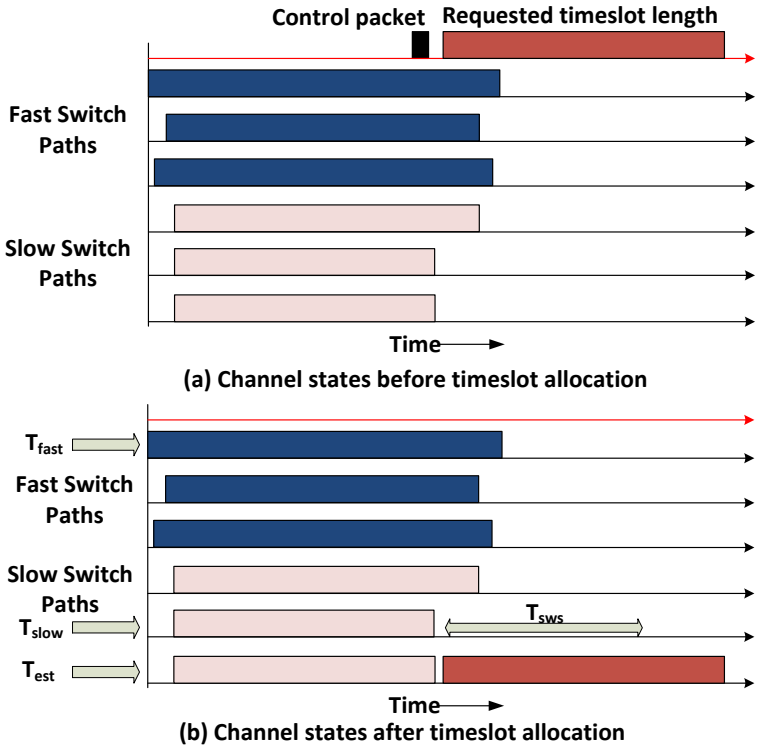


Figure 3.7. Timeslot Allocation for HOSA: Case 3

Figure 3.7 presents another scenario when there is an already established slow switch path exists between ToR pairs. In this scenario, the incoming burst is scheduled only on the already established slow path i.e. 3rd channel of the slow switch paths because the latency will be lower in this case.

Figure 3.8 represents a new scenario in which a new slow switch path is established, although an already established slow switch path exists between ToR pairs. In this scenario, timeslots are assigned to a fast switch path and in a new slow switch

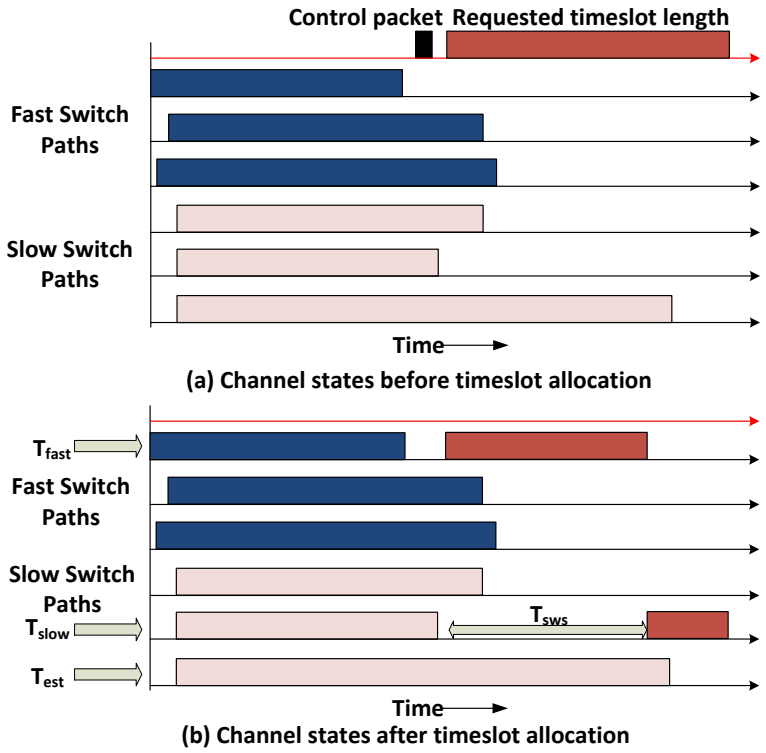


Figure 3.8. Timeslot Allocation for HOSA: Case 4

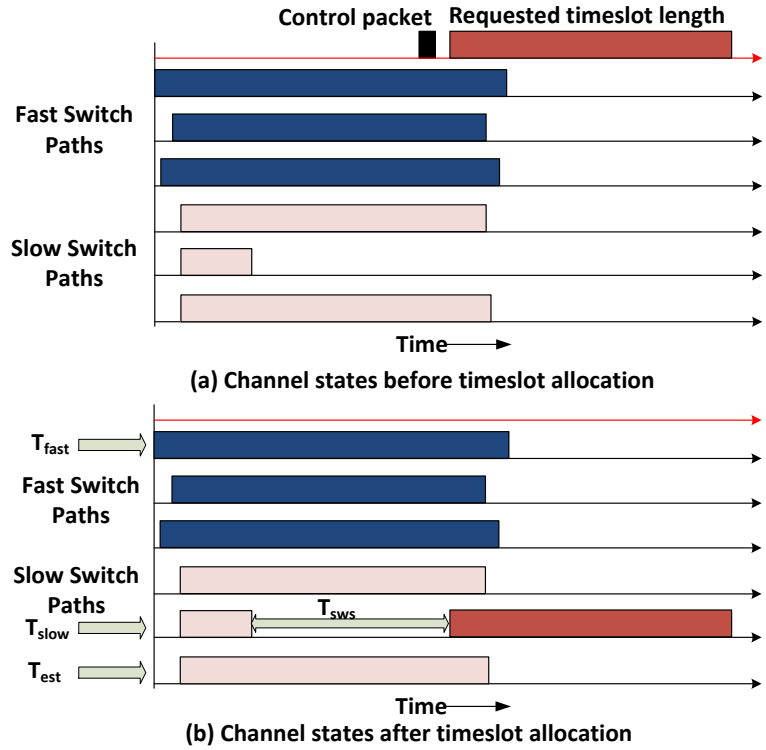


Figure 3.9. Timeslot Allocation for HOSA: Case 5

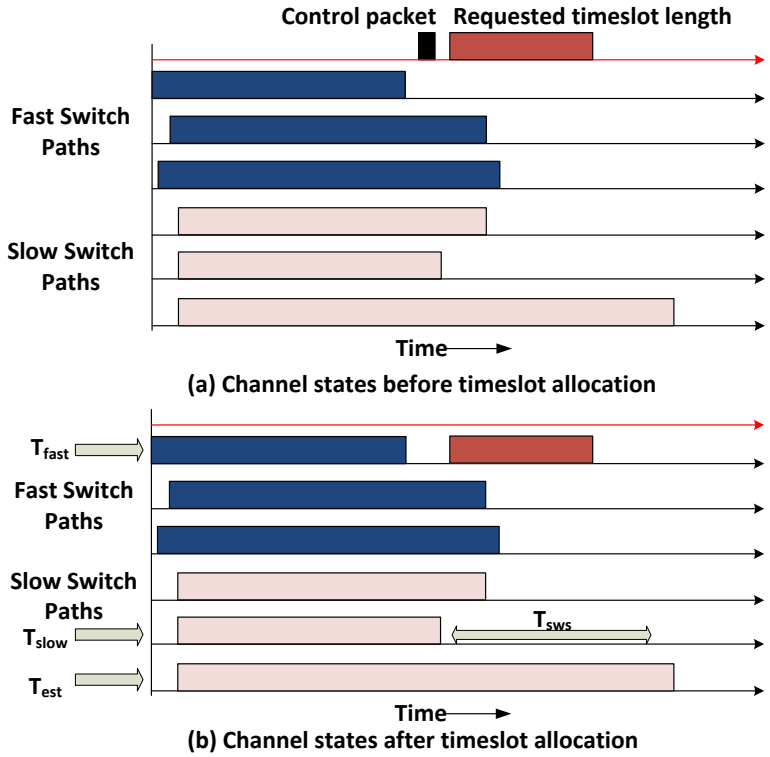


Figure 3.10. Timeslot Allocation for HOSA: Case 6

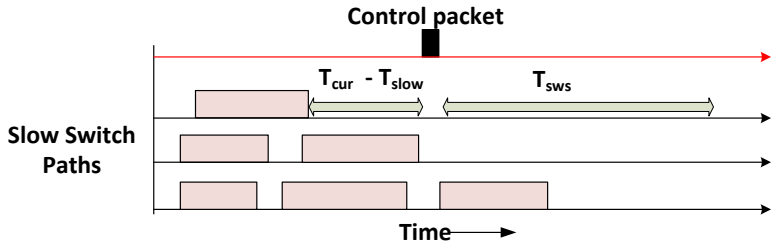


Figure 3.11. Establishment of a new slow switch path using speculation approach.

path i.e. first channel of the fast switch paths and 2nd channel of the slow switch paths.

Figure 3.9 shows a scenario when an incoming burst is assigned a timeslot in a slow switch path by establishing a new connection i.e. 2nd channel in the slow switch paths. Similarly Figure 3.10 shows a scenario when an incoming burst is assigned to a timeslot in a fast switch path. In this way low latency is achieved in both cases.

Figures 3.11 represents a mechanism for the establishment of a new slow switch path using the speculation approach. This path is established for future requests. A new slow switch path will be established if the condition $T_{cur} - T_{slow} \geq T_{idle}$ is satisfied.

3.2.7 Switch Configuration

Switch configuration is the final operation of the controller. After processing the control packet, a configuration message is generated and is sent to the switch controller to configure the optical switch. The configuration message contains the source, the destination port numbers and the connection start-time. It does not contain the connection end-time because connections are established with unlimited duration. The connection end-time information is only maintained by the controller. The switch controller configures the optical switch according to the instructions in the configuration message.

The biggest advantage of establishing a connection with an unlimited duration is that when a new connection request arrives at the controller for an already established connection in the slow path, the controller only updates its horizon with a new time and nothing is done in the switch controller. However, in order to ensure fairness, this principle does not apply on the fast switch path. In the fast switch path, all the traffic has an equal probability of getting a timeslot while in slow switch paths, the probability of being assigned a timeslot on an already established connection is higher than of a timeslot on an yet unestablished connection. This is to avoid frequent reconfiguration of MEMS switches so that persistent traffic flows are routed through the slow switch paths.

3.3 Scalability Analysis of HOSA

According to the assumptions made in Section 3.2.1, this thesis considers equal size of fast and slow optical switch i.e. $N \times N$ in N rack DCN. So, scalability analysis has been done by considering equal size of fast and slow optical switch. Slow optical MEMS switches with 320 bidirectional ports are commercially available while fast optical switches can be built using technologies described in Chapter 2. Due to the constraint of maximum port size of fast or slow optical switches, the scalability of the architecture with only one fast and one slow switch is limited to a few thousand servers as shown in the first two rows of Table 3.1. This is suitable for departmental

and medium-scale enterprise data centres, but this research targets large-scale high performance computing data centres in the order of O(10K) of servers. In order to achieve this, multiple optical fast and slow switches in the core arranged in a single stage topology are employed as shown in Figure 3.1.

Table 3.1 describes different configurations of slow and fast switches (SS and FS respectively), ratio of their capacities, number of slow and fast switches required (N_{SS} and N_{FS} respectively), servers per rack (S_{RK}), and various core to edge oversubscription ratios (C_O). It can be seen that maximum system size with only one fast and one slow switch, each having [320 × 320] configurations and oversubscription ratios of 4 and 2 is limited to 2560 and 1280 servers respectively. The maximum size of the system reaches to 12800 servers having 40 servers per rack and [320 × 320] switches configuration with different capacities of slow and fast switches. It can be observed that the number of slow and fast optical switches required are varied with the capacity of slow:fast switches. Slow MEMS switches with 1024 ports are feasible [26, 102] and fast optical switch using SOAs with 1024 ports has also been proposed [29]. The system size of 40960 servers with 40 servers per rack and 81920 servers with 80 servers per rack can be achieved with the proposed single stage topology without converting to multi-stage core topologies. 80 servers per rack can be integrated by using two 64 port ToR switches per rack. Similarly, if a pod switch is considered instead of the ToR switch that has the capacity to integrate several ToR switches into a single unit and can aggregate a few hundreds to thousand servers [23], leads to the scalability upto 245760 servers by considering 240 servers per pod which is ideal for future large scale data centres.

It can be observed that the size of the optical switch (port density) controls the maximum number of racks while the number of optical switches controls the core oversubscription ratio. Single-stage core interconnect topology with multiple optical switches allows proposed design to both incrementally scaled up (in capacity) and scaled out (in the number of racks) without requiring major re-cabling and network re-configuration similar to the topology used in reconfigurable architecture [72].

The multi-stage core topology is avoided due to the complexity of the control plane

Table 3.1. Scalability Analysis of HOSA

SS	FS	$SS:FS$	N_{SS}	N_{FS}	S_{RK}	C_O	T_{RK}	$Servers$
[320 × 320]	[320 × 320]	50 : 50	1	1	40	4	64	2560
			1	1	40	2	32	1280
[320 × 320]	[320 × 320]	50 : 50	5	5	40	4	320	12800
			10	10	40	2	320	12800
		60 : 40	6	4	40	4	320	12800
			12	8	40	2	320	12800
		70 : 30	7	3	40	4	320	12800
			14	6	40	2	320	12800
		80 : 20	8	2	40	4	320	12800
			16	4	40	2	320	12800
[1024 × 1024]	[1024 × 1024]	50 : 50	5	5	40	4	1024	40960
			10	10	40	2	1024	40960
		50 : 50	10	10	80	4	2048	81920
			20	20	80	2	2048	81920
[1024 × 1024]	[1024 × 1024]	50 : 50	30	30	240	4	6144	245760
			60	60	240	2	6144	245760

and optical signal degradation at every intermediate optical switch (providing all optical switching) due to insertion losses and crosstalk. Optical amplifiers may be required in multi-stage core topologies that will not only increase the overall cost of the interconnect but also the power consumption. The multi-stage designs can be scaled to a very large topology but scaling is expensive and is not incremental.

3.4 Cost and Power Consumption Analysis

In the analysis for the cost and power consumption, the cost and power consumption of the network elements (i.e. transceivers and switches) that are used in the interconnection network are considered. The cost and power consumption of cooling, cabling and other elements are not considered. Table 3.2 shows the cost and power consumption of different network elements. Apart from the MEMS and AWGR, the cost of all other devices/components are available online in the references mentioned with them

Table 3.2. Cost and Power Consumption of Network Elements

Element	Symbols	Cost[$]	Power[W]
ToR Switch	$C_{TOR_{CMOS}}/P_{TOR_{CMOS}}$	438 /port [105]	3.7/port [16]
Aggregate/core Switch	$C_{Core_{CMOS}}/P_{Core_{CMOS}}$	875 /port [106]	12.5/port [17]
MEMS	C_M/P_M	500/port [26]	0.14/port [107]
10G Trans	C_{TR}/P_{TR}	400 [108]	1.5 [109]
AWGR	C_{AWGR}	250/port [61]	None
Tunable Lasers	C_{TL}/P_{TL}	1540 [110]	2 [111]
FPGA	C_{FPGA}/P_{FPGA}	903 [112]	168 [113]

while for the MEMS and AWGR, the cost is selected by getting quotations from vendors. The value of power consumption of these devices is selected from the datasheets. The datasheets of all the devices are available online as mentioned in the references. However, the datasheets for some of the devices such as ToR, aggregate/core switch, MEMS and AWGR are also provided in Appendix A.

Four interconnection networks are considered in a comparative analysis with HOSA. These networks are Fat Tree, BCube, Traditional-electrical (TE) and Optical-electrical (OE). Various capacities of fast and slow switches in the HOSA are considered in the comparison with these four networks. The proposed design is scalable to 40960 and 81920 servers using ToR switches at the edge as described in section 3.3, and so these two values for servers are used to compare the proposed design with other networks.

3.4.1 Fat Tree Network

The Fat Tree (FT) is the most common tree topology used in DCNs. The FT with *n-port* switches can connect $\frac{n^3}{4}$ hosts/servers with a total number of $\frac{5n^3}{4}$ switch ports [114]. The power consumption of the FT network P_{FT} is calculated using the following

formula:

$$\begin{aligned} P_{FT} = &\ Total\ power\ consumption\ of\ switch\ ports\ with\ transceivers + \\ &\ total\ power\ consumption\ of\ transceivers\ in\ servers \\ P_{FT} = &\ \frac{5n^3}{4}(P_{TOR_{CMOS}} + P_{TR}) + \frac{n^3}{4}P_{TR} \end{aligned} \tag{3.2}$$

where $P_{TOR_{CMOS}}$ is the power consumption of ToR switch port and P_{TR} is the power consumption of an optical transceiver. Since all the switches in the FT network are of the same size, so only ToR switches are considered in FT network. As, the FT network have $\frac{n^3}{4}$ servers, so we cannot have exactly 40960 servers according to this formula. In this case, the nearest value to 40960 servers can be considered in the FT network. For example, 54 port switches result in $\frac{54^3}{4} = 39366$ servers while 56 port switches result in $\frac{56^3}{4} = 43904$ servers. In this case, the nearest value is 39366 with 54 port switches. The total number of switch ports for 54 ports switches are $\frac{5\times54^3}{4}$. The power consumption of the FT network with 39366 servers is calculated and then it is normalized to 40960 servers. So the power consumption of the FT network with 39366 servers is given by:

$$P_{FT} = \frac{5 \times 54^3}{4}(3.7 + 1.5) + \frac{54^3}{4} \times 1.5 = 1.082565\ MW \tag{3.3}$$

The normalized power consumption $P_{FT}N$ of the FT network with 40960 servers is given by:

$$\begin{aligned} P_{FT}N &= \frac{P_{FT}}{Nearest\ value}(Actual\ value) \\ P_{FT}N &= \frac{1.082565}{39366}(40960) \ = \ 1.1264\ MW \end{aligned} \tag{3.4}$$

A similar approach is taken to estimate the power consumption of 81920 servers.

The CAPEX cost of the FT network C_{FT}^{CAPEX} is calculated using the following formula:

$$\begin{aligned} C_{FT}^{CAPEX} = &\ Total\ cost\ of\ switch\ ports\ with\ transceivers + \\ &\ total\ cost\ of\ transceivers\ in\ servers \\ C_{FT}^{CAPEX} = &\ \frac{5n^3}{4}(C_{TOR_{CMOS}} + C_{TR}) + \frac{n^3}{4}C_{TR} \end{aligned} \tag{3.5}$$

where $C_{TOR_{CMOS}}$ is the cost of ToR switch port and C_{TR} is the cost of the transceiver. Similarly, the CAPEX cost of the FT network with 39366 servers is given by:

$$C_{FT}^{CAPEX} = \frac{5 \times 54^3}{4}(438 + 400) + \frac{54^3}{4} \times 400 \\ = 180.68994\, M\ US\$ \tag{3.6}$$

The normalized cost $C_{FT}^{CAPEX}N$ of the FT network with 40960 servers is given by:

$$C_{FT}^{CAPEX}N = \frac{C_{FT}^{CAPEX}}{Nearest\ value}(Actual\ value) \\ C_{FT}^{CAPEX}N = \frac{180.68994}{39366}(40960)\ =\ 188.0064\ \ M\ US\$ \tag{3.7}$$

A similar approach is taken to estimate the CAPEX cost of 81920 servers. The OPEX cost is related to the power consumption. In order to calculate OPEX cost, a value of 10 cent per unit cost of electricity is considered which is the average cost per unit of electricity in the United States [115]. The OPEX cost of the FT network C_{FT}^{OPEX} is calculated using the following formula:

$$C_{FT}^{OPEX} = P_{FT} \times 1000 \times 24 \times 365 \times 0.1 \times Years \tag{3.8}$$

3.4.2 BCube Network

BCube network is a non-tree based architecture proposed for use in modular data centres (MDCs) [116]. The BCube network takes a server-centric approach, rather than a switch-oriented approach. Servers have multiple ports that are connected with the different levels of electrical switches. Servers not only send their traffic to other servers but they also work as switches to forward traffic from other servers. The $BCube_0$ has n servers which are connected with an n-port switch. The $BCube_k$ ($k \geq 1$) is constructed from n $BCube_{k-1}s$ and n^k n-port switches. There are $k+1$ level of switches and each level has n^k n-port switches. There is a total of $N = n^{k+1}$ servers and each server has $k+1$ ports. Similar to FT network, the $BCube$ network also has equal size of electrical switches. So, ToR switches are considered for this network as well. The

power consumption of the $BCube_k$ network P_{BCube} is calculated using the following formula:

$$\begin{aligned} P_{BCube} = &\ Total\ power\ consumption\ of\ switch\ ports\ with\ transceivers + \\ &\ total\ power\ consumption\ of\ transceivers\ in\ servers \\ P_{BCube} = &\ n^k(k+1).n(P_{TOR_{CMOS}} + P_{TR}) + n^{k+1}(k+1)(P_{TR}) \end{aligned} \tag{3.9}$$

This thesis considers a $BCube_3$ network with $k = 3, n = 14$ having $N = 14^{3+1} = 38416$ servers and each server has $k+1 = 4$ ports. So the power consumption of this network is given by:

$$\begin{aligned} P_{BCube_3} &= 14^3(3+1) \times 14(3.7+1.5) + 14^{3+1}(3+1)(1.5) \\ &= 1.0295488\ \ MW \end{aligned} \tag{3.10}$$

Similar to the Fat Tree networks, the BCube networks also cannot have exactly 40960 servers, so this value of power consumption is normalized to 40960 servers which is given by:

$$\begin{aligned} P_{BCube}N &= \frac{P_{BCube}}{Nearest\ value}(Actual\ value) \\ P_{BCube}N &= \frac{1.0295488}{38416}(40960)\ =\ 1.097728\ MW \end{aligned} \tag{3.11}$$

The CAPEX cost of the $BCube_k$ network C^{CAPEX}_{BCube} is given by:

$$\begin{aligned} C^{CAPEX}_{BCube} = &\ Total\ cost\ of\ switch\ ports\ with\ transceivers + \\ &\ total\ cost\ of\ transceivers\ in\ servers \\ C^{CAPEX}_{BCube} = &\ n^k(k+1).n(C_{TOR_{CMOS}} + C_{TR}) + n^{k+1}(k+1)(C_{TR}) \end{aligned} \tag{3.12}$$

Similarly, the CAPEX cost of the BCube network having 38416 servers is given by:

$$\begin{aligned} C^{CAPEX}_{BCube} &= 14^3(3+1) \times 14(438+400) + \\ 14^{3+1}(3+1) &\times 400 = 190.236032\ \ M\ US\ \$ \end{aligned} \tag{3.13}$$

For 40960 servers, the normalized cost is calculated by:

$$C_{BCube}^{CAPEX}N = \frac{C_{BCube}^{CAPEX}}{Nearest\ value}(Actual\ value)$$
$$C_{BCube}^{CAPEX}N = \frac{190.236032}{38416}(40960)\ =\ 202.83392\ \ M\ US\ \$ \tag{3.14}$$

The OPEX cost of the BCube network C_{BCube}^{OPEX} is calculated using the following formula:

$$C_{BCube}^{OPEX} = P_{BCube} \times 1000 \times 24 \times 365 \times 0.1 \times Years \tag{3.15}$$

3.4.3 Traditional Electrical Network

For the traditional electrical (TE) network, this thesis considers a two layer topology having edge/pod and core switches. The edge/pod switch is considered to be a cluster of the ToR switches making an aggregation layer. The power consumption of the TE network P_{TE} is calculated using the following formula:

$$P_{TE} = P_{EDGE}^{TE} + P_{CORE}^{TE} + T_{SR}.P_{TR} \tag{3.16}$$

where P_{EDGE}^{TE} and P_{CORE}^{TE} represent the total power consumption at the edge and the core switches respectively. T_{SR} is the total number of servers in the network. P_{EDGE}^{TE} is calculated using the following formula:

$$P_{EDGE}^{TE} = T_{RK}(2 \times S_{RK} + N_A)(P_{TOR_{CMOS}} + P_{TR}) \tag{3.17}$$

where T_{RK} represents the total number of racks in the network and S_{RK} denotes the number of servers in the rack. The S_{RK} transceivers in the ToR switch are connected to the servers in the rack and remaining S_{RK} transceivers are connected to the core switches. N_A is the number of ports per ToR switch connecting other ToR switches to make the pod switch. The P_{CORE}^{TE} is calculated using the following formula:

$$P_{CORE}^{TE} = (T_{RK} \times S_{RK})(P_{CORE_{CMOS}} + P_{TR}) \tag{3.18}$$

For example, the power consumption P_{TE} for 40960 servers is calculated by substituting appropriate values as shown below:

$$P^{TE} = 1024(2 \times 40 + 15)(3.7 + 1.5) + (1024 \times 40)(12.5 + 1.5) + (40960 \times 1.5) = 1.087890625 \ MW \tag{3.19}$$

Similarly, the CAPEX cost of the TE network C_{TE}^{CAPEX} is calculated by using formula:

$$C_{TE}^{CAPEX} = C_{EDGE}^{TE} + C_{CORE}^{TE} + T_{SR}.C_{TR} \tag{3.20}$$

where C_{EDGE}^{TE} and C_{CORE}^{TE} represent the total cost of the edge and the core switches respectively. C_{EDGE}^{TE} and C_{CORE}^{TE} are calculated using the following formulae:

$$C_{EDGE}^{TE} = T_{RK}(2 \times S_{RK} + N_A)(C_{TOR_{CMOS}} + C_{TR}) \tag{3.21}$$

$$C_{CORE}^{TE} = (T_{RK} \times S_{RK})(C_{CORE_{CMOS}} + C_{TR}) \tag{3.22}$$

For example, the CAPEX cost C_{TE}^{CAPEX} for 40960 servers is calculated by substituting appropriate values as shown below:

$$C_{TE}^{CAPEX} = 1024(2 \times 40 + 15)(438 + 400) + (1024 \times 40)(875 + 400) + 40960 \times 400 = 150.12864 \ M\ US\$ \tag{3.23}$$

The OPEX cost of the TE network C_{TE}^{OPEX} is calculated by using the following formula:

$$C_{TE}^{OPEX} = P_{TE} \times 1000 \times 24 \times 365 \times 0.1 \times Years \tag{3.24}$$

3.4.4 Optical/Electrical Network

An abstract model for the optical/electrical network similar to Helios [23] is considered in the analysis. It consists of two layer of switches: edge/pod and core. The edge switches are clusters of ToR switches while the core switches are a combination of electrical and MEMS switches. The core layer is fully subscribed in which half of the links are connected to the electrical switches and other half to the MEMS switches. The

power consumption of the OE network P_{OE} is calculated by using following formula:

$$P_{OE} = P_{EDGE}^{OE} + P_{CORE}^{OE} + T_{SR}.P_{TR} + P_{CP}^{OE} \tag{3.25}$$

where P_{EDGE}^{OE} and P_{CORE}^{OE} represent the total power consumption at the edge and the core switches respectively while P_{CP}^{OE} is the total power consumption of the control plane. P_{EDGE}^{OE} and P_{CORE}^{OE} are calculated by using the following formulae:

$$P_{EDGE}^{OE} = T_{RK}(2 \times S_{RK} + N_A + N_{CP})(P_{TOR_{CMOS}} + P_{TR}) \tag{3.26}$$

$$P_{CORE}^{OE} = (T_{RK}.\frac{S_{RK}}{2})(P_{CORE_{CMOS}} + P_{TR}) + (T_{RK}.\frac{S_{RK}}{2})(P_M) \tag{3.27}$$

where N_{CP} is the number of transceivers in each ToR switch dedicated to the control plane and $\frac{S_{RK}}{2}$ is the number of links in the electrical core and the MEMS switches which are connected with the pod switches. P_M is the power consumption of MEMS switch. P_M not only includes the power consumption of a single port of MEMS switch but it also includes the power consumption of the MEMS switch controller. The MEMS switch controller is a built-in controller that comes with the MEMS switch and is used to configure MEMS switch. The P_{CP}^{OE} is calculated using following formula:

$$P_{CP}^{OE} = T_{RK}(P_{TOR_{CMOS}} + P_{TR}) + N_{TR_{Cont}}.P_{TR} \tag{3.28}$$

where $N_{TR_{Cont}}$ is the number of transceivers in the controller server. A single centralized controller is considered in this model. For example, the power consumption P_{OE} for 40960 servers is calculated by substituting appropriate values from Table 3.2 as shown below:

$$\begin{aligned} P_{OE} = 1024(2 \times 40 + 15 + 1)(3.7 + 1.5) + \\ (1024 \times \frac{40}{2})(12.5 + 1.5) + (1024 \times \frac{40}{2})(0.14) + 40960 \times 1.5 + \\ 1024(3.7 + 1.5) + 1 \times 1.5 = 0.823731899 \ MW \end{aligned} \tag{3.29}$$

A similar approach is used to calculate CAPEX cost of the OE network. The CAPEX cost of the OE network C_{OE}^{CAPEX} is given by:

$$C_{OE}^{CAPEX} = C_{EDGE}^{OE} + C_{CORE}^{OE} + T_{SR}.C_{TR} + C_{CP}^{OE} \tag{3.30}$$

where C_{EDGE}^{OE} and C_{CORE}^{OE} represent the total cost of the edge and the core switches and C_{CP}^{OE} is the total cost of the control plane. The C_{EDGE}^{OE}, C_{CORE}^{OE} and C_{CP}^{OE} values are calculated by using the following formulae:

$$C_{EDGE}^{OE} = T_{RK}(2 \times S_{RK} + N_A + N_{CP})(C_{TOR_{CMOS}} + C_{TR}) \tag{3.31}$$

$$C_{CORE}^{OE} = (T_{RK}.\frac{S_{RK}}{2})(C_{CORE_{CMOS}} + C_{TR}) + (T_{RK}.\frac{S_{RK}}{2})(C_M) \tag{3.32}$$

$$C_{CP}^{OE} = T_{RK}(C_{TOR_{CMOS}} + C_{TR}) + N_{TR_{Cont}}.C_{TR} \tag{3.33}$$

where C_M is the cost of the MEMS switch port. For example, the CAPEX cost C_{OE} for 40960 servers is calculated by substituting appropriate values from Table 3.2 as shown below:

$$\begin{aligned} C_{OE}^{CAPEX} &= 1024(2 \times 40 + 15 + 1)(438 + 400) + \\ (\frac{1024}{2} \times 40)(875 + 400) + (\frac{1024}{2} \times 40)(500) &+ 40960(400) \\ +1024(438 + 400) + 1 &\times 400 \\ = 135.524752 \ \ M\ US\$ & \end{aligned} \tag{3.34}$$

The OPEX cost of the OE network C_{OE}^{OPEX} is calculated by using the following formula:

$$C_{OE}^{OPEX} = P_{OE} \times 1000 \times 24 \times 365 \times 0.1 \times Years \tag{3.35}$$

3.4.5 HOSA

The proposed design HOSA also consists of two layer of switches: ToR switches at the edge and an array of optical switches at the core. As with the TE and OE networks, the core layer is fully subscribed.

The power consumption of the HOSA (P_{HOSA}) is calculated by using following formula:

$$P_{HOSA} = P_{EDGE}^{HOSA} + P_{CORE}^{HOSA} + T_{SR}.P_{TR} + P_{CP}^{HOSA} \tag{3.36}$$

where P_{EDGE}^{HOSA} and P_{CORE}^{HOSA} represent total power consumption at the edge and the core switches respectively while P_{CP}^{HOSA} is the total power consumption of the control plane. The P_{EDGE}^{HOSA} and P_{CORE}^{HOSA} are calculated by using the following formulae:

$$P_{EDGE}^{HOSA} = T_{RK}(2 \times S_{RK} + N_{CP})(P_{TOR_{CMOS}} + P_{TR}) \tag{3.37}$$

$$P_{CORE}^{HOSA} = T_{RK}.NS_{RK}.P_M + T_{RK}.NF_{RK}.P_F \tag{3.38}$$

where NS_{RK} and NF_{RK} represent the number of transceivers per ToR switch connected with the slow and fast switches respectively while P_M and P_F are the power consumption per port of the MEMS and the fast optical switches respectively. The P_{CP}^{HOSA} is calculated using the following formula:

$$P_{CP}^{HOSA} = T_{RK}(P_{TOR_{CMOS}} + P_{TR}) + N_{TR_{Cont}}.P_{TR} + P_{FPGA} \tag{3.39}$$

where $N_{TR_{Cont}}$ is the number of transceivers in the controller server that performs routing and scheduling algorithm. This research is based on a single centralized controller. Additionally, this centralized controller is assumed to have an $FPGA$ card to implement routing and scheduling algorithm in the hardware domain for efficient processing. AWGR switch is considered as a reference design for the fast optical switch because it is commercially available while SOA-based switches in broadcast and select architecture are not available commercially. The power consumption P_{HOSA} for a configuration of 40960 servers with $FS = 0.5$ and $SS = 0.5$ (where FS represents the capacity of the fast switches while SS represents the capacity of the slow switches) is

calculated by substituting appropriate values from Table 3.2 as shown below:

$$\begin{aligned} P_{HOSA} = 1024(2 \times 40 + 1)(3.7 + 1.5) + \\ 1024 \times 10 \times 0.14 + 1024 \times 10 \times 4 + 40960 \times 1.5 \\ 1024(3.7 + 1.5) + 1 \times 168 + 1 \times 1.5 \\ = 0.552407742\, MW \end{aligned} \tag{3.40}$$

The CAPEX cost of the HOSA C_{HOSA}^{CAPEX} is calculated by using the same method as used for the calculation of power consumption. The C_{HOSA}^{CAPEX} is calculated by using the following formula:

$$C_{HOSA}^{CAPEX} = C_{EDGE}^{HOSA} + C_{CORE}^{HOSA} + T_{SR}.C_{TR} + C_{CP}^{HOSA} \tag{3.41}$$

where C_{EDGE}^{HOSA} and C_{CORE}^{HOSA} are the total cost of the edge and the core switches respectively while C_{CP}^{HOSA} represents the cost of the control plane. The C_{EDGE}^{HOSA} and C_{CORE}^{HOSA} are calculated by using the following formulae:

$$C_{EDGE}^{HOSA} = T_{RK}(2 \times S_{RK} + N_{CP})(C_{TOR_{CMOS}} + C_{TR}) \tag{3.42}$$

$$C_{CORE}^{HOSA} = T_{RK}.NS_{RK}.C_M + T_{RK}.NF_{RK}.C_F \tag{3.43}$$

where C_M and C_F represent the cost per port of the MEMS and the fast optical switch respectively. The C_{CP}^{HOSA} is calculated by using the following formula:

$$C_{CP}^{HOSA} = T_{RK}(C_{TOR_{CMOS}} + C_{TR}) + N_{TR_{Cont}}.C_{TR} + C_{FPGA} \tag{3.44}$$

The cost of MEMS switch is 500$ per port while for fast optical switch, I conservatively selected high value for cost i.e. 4000$ per port. However, its actual cost will be 3330$. The CAPEX cost C_{HOSA} for 40960 servers with $FS = 0.5$ and $SS = 0.5$ is calculated

as shown below:

$$\begin{aligned} C_{HOSA}^{CAPEX} &= 1024(2\times 40+1)(438+400)+ \\ 1024\times 20\times 500 &+ 1024\times 20\times 4000+40960\times 1 \\ 1024(438+400) &+ 1\times 903+1\times 400 \\ &= 178.910487 \ \ M\ US\$ \end{aligned} \tag{3.45}$$

The OPEX cost of the HOSA C_{HOSA}^{OPEX} is calculated by using the following formula:

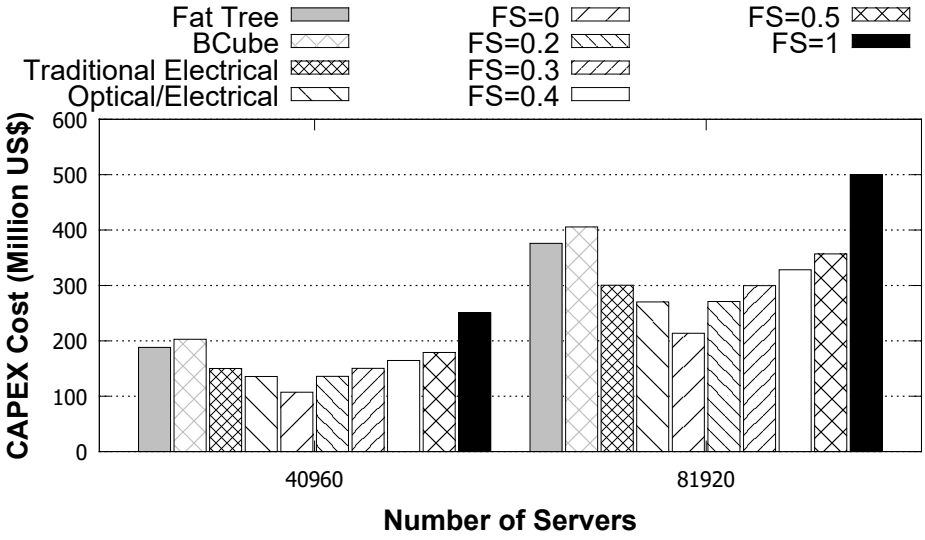
$$C_{HOSA}^{OPEX} = P_{HOSA}\times 1000\times 24\times 365\times 0.1\times Years \tag{3.46}$$

3.4.6 Results

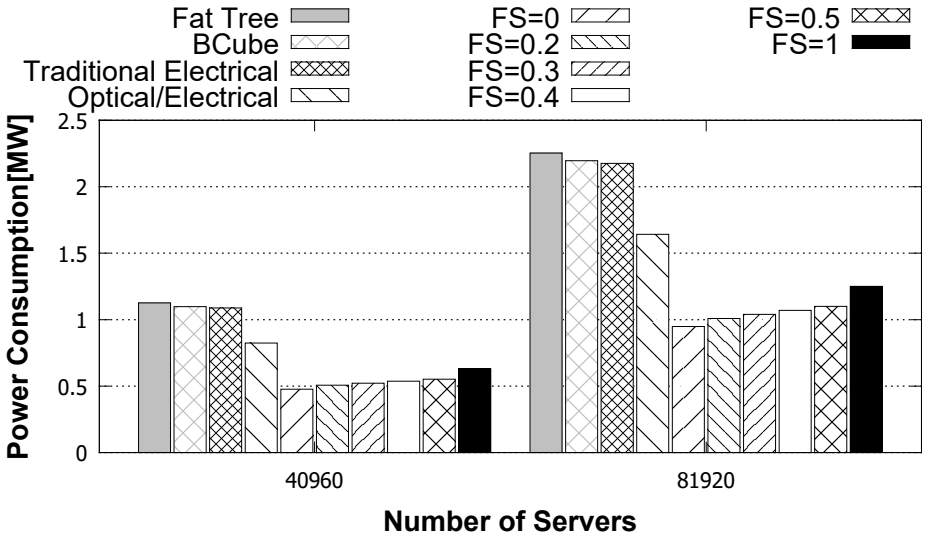
In this analysis, the CAPEX cost, OPEX cost and power consumption of the FT, BCube, TE, OE and HOSA networks are investigated and the results are shown in Figures 3.12 and 3.13.

The cost and power consumption of the HOSA are calculated using different capacities of the fast and slow optical switches. In Figures 3.12(a), 3.12(b) and 3.13, FS represents the capacity of the fast optical switches. $FS = 0$ means that there is no fast optical switch and all of the switching capacity is provided by the slow optical switches while $FS = 1$ indicates that there is no slow optical switch and all of the switching capacity is provided by the fast optical switches. These are the two extreme cases, which are considered as the worst and the best case respectively. Similarly, $FS = 0.2$ means that 20% of the switching capacity is provided by the fast optical switches and the remaining 80% capacity is provided by the slow optical switches. Same approach is used for other values of FS. The various combinations of the switching capacities in HOSA are compared with two extreme cases as well as with other networks.

Figure 3.12(a) shows that the CAPEX cost of the interconnection network using only fast optical switches is 20-30% higher as that of the BCube and FT network while it is almost two third to that of TE or OE networks, but this cost is reduced by one third by considering HOSA with $FS = 0.4$. The cost of HOSA with this combination is almost



(a)



(b)

Figure 3.12. Total CAPEX cost and power consumption of different interconnection networks with respect to various values for the number of servers, (a) CAPEX Cost, (b) Power Consumption

the same as that of TE network but it is slightly higher than OE network. This extra upfront cost is mitigated to some extent by its reduced OPEX cost as shown in Figure 3.13. The reduced OPEX cost results from the improvement in power consumption as shown in Figure 3.12(b).

It can be seen from Figure 3.13 that with different capacities of the HOSA, almost 50% improvement in power consumption is achieved over the FT, BCube and TE net-

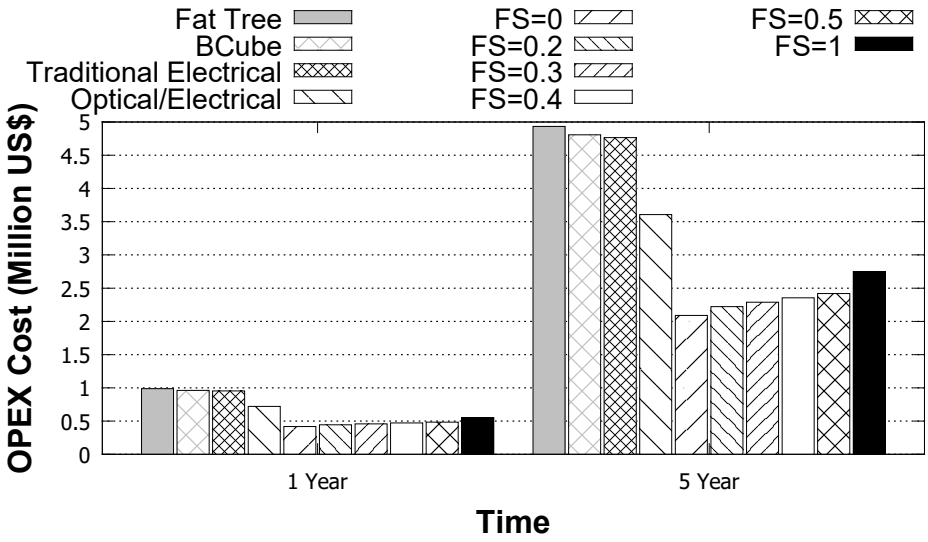


Figure 3.13. Total OPEX cost of different interconnection networks with respect to years using 40960 servers.

works while a $30-35\%$ improvement in power consumption is achieved over the OE network.

3.5 Performance Analysis

To assess the performance of the architecture described above, simulations models have been developed using the OMNeT++ simulation framework [117]. OMNeT++ is an open source object-oriented modular discrete event network simulation framework. It has a generic architecture, so it can be (and has been) used in various problem domains such as modelling of wired and wireless communication networks, protocol modelling, modelling of queueing networks and in modelling/simulation of any system where the discrete event approach is suitable [117]. Furthermore, OMNeT++ has a model library called INET Framework that provides protocols, agents and other models for researchers working with communication networks. For example, INET contains models for the Internet stack (TCP, UDP, IPv4, IPv6, OSPF, BGP, etc.), wired and wireless link layer protocols (Ethernet, PPP, IEEE 802.11, etc), support for mobility, MANET protocols, several application models, and many other protocols and components [118].

Due to the modular architecture of the OMNeT++ and its INET framework that provides a huge library of protocols/components, this research used OMNeT++ to simulate and validate the proposed architectures. The developed simulation models contain models for ToR switches, fast/slow optical switches and the controller while OMNeT++ INET models for servers and electrical switches are used. The ToR switch models implement the full logic of the control packet and the burst generation algorithms. The controller model implements logic of the control plane such as routing, scheduling and switch configuration while the fast/slow switch models implement the logic of switch configuration and all-optical data forwarding.

In this section, the simulation model of the HOSA, together with important simulation parameters, traffic generation and simulation scenarios is discussed.

3.5.1 Simplified Model of HOSA

The key simulation parameters are listed in Table 3.3. The simplified model consists of $N = 20$ ToR switches. Each ToR switch has $H = 40$ servers connected to it via 10 Gbps Ethernet links. Each ToR switch is connected to an electrical switch via a 10 Gbps Ethernet link which is used for the control plane. Only one electrical switch is required for the control plane which is connected to the controller, all ToR and all of the optical switches. Two optical switches are considered, i.e. one for the slow path and other for the fast path. Each ToR switch also has $X = 20$ optical transceivers operating at 10 Gbps, of which $K = 10$ links are connected to the slow optical switch and $X - K = 10$ links are connected to the fast optical switch.

3.5.2 Traffic Generation

To the best of the author's knowledge, no theoretical model or benchmark of data centre traffic has found acceptance in the research community yet, but some studies [102, 119–122] have investigated the nature of the data centre traffic. Traffic in data centres is bursty in nature and shows evidence of ON-OFF behaviour [121, 122].

Table 3.3. Simulation Parameters for HOSA

Parameter Name	Symbol	Value
Racks/ToR Switches	N	20
Servers per rack	H	40
Fast & Slow Switch		1 Each
Electrical Switches (Management Network)		1
Radix/Degree of ToR Switches	X	20
Optical links to slow switch from ToR switch	K	10
Optical links to fast switch from ToR switch	$X-K$	10
Data rate		10 Gbps per link
Network over-subscription Ratio		2:1
Switching Time of Slow Switch	T_{sws}	10ms
Switching Time Fast Switch	T_{swf}	$1\mu s$
Control packet processing time	T_{proc}	$50\mu s$
Overhead	T_{oh}	$1\mu s$
Burst Aggregation Time	T_a	{1ms,2ms}
ON Period Length		Exponential(10ms)
OFF Period Length		Exponential(2ms)
Topological Degree of Communication	TDC	{1,4,8}

In order to model bursty traffic, several distributions can be used such as Interrupted Poisson Process (IPP), Lognormal, Weibull, Markov chain and Markov Modulated Poisson Process etc [121,123]. In this work, Markov chain and Weibull distributions are used to generate bursty traffic. In HOSA and HOSA with TDS, Markov chain is considered while in FOSA, Weibull distribution is used to generate traffic.

Markov models are used to model the activities of a traffic source on a network, by a finite number of states. The complexity of the model increases proportionally with increasing number of states. An important aspect of the Markov model - the Markov Property, states that the next (future) state depends only on the current state. In other words the probability of the next state, denoted by some random variable $X_n + 1$, depends only on the current state, indicated by X_n, and not on any other state X_i, where $i < n$. [123]. Markov chain is a stochastic process that satisfies the Markov property (usually characterized as "memorylessness") [124]. For example, lets suppose a set of states, $S = s_1, s_2, ..., s_r$. The process starts in one of these states and moves successively from one state to another. If the chain is currently in state

s_i, then it moves to state s_j at the next step with a probability denoted by p_{ij} , and this probability does not depend upon which state the chain was in before the current state. The probabilities p_{ij} are called transition probabilities. The process can remain in the state it is in, and this occurs with probability p_{ii} [125].

In this chapter, Markov chain with two states are considered, i.e. ON state and OFF state. Packets are generated during the ON state with exponential arrivals while in the OFF state, no packet is generated. The ON and OFF periods are independent and exponentially distributed. Various exponential inter-arrival rates of packets are considered during the ON period to investigate traffic at different loads. Markov chain with only two states is easy to implement. This is because, it is widely used to model the bursty traffic in optical burst swtiching [44, 126–129].

The INET library of the OMNeT++ provides built in functionality of generating ON/OFF bursty traffic using various traffic models. UDP traffic is used to model bursty traffic using ON/OFF periods in OMNeT++. This research evaluates the performance of the system using both UDP and TCP traffic. In chapter 3-5, the performance of the system has been evaluated using UDP traffic, while in chapter 6 the performance evaluation of the system using TCP traffic has also been done.

Traffic generation is also controlled by the Topological Degree of Communication (TDC) which is the number of simultaneous destinations ToR switches that a given ToR switch sends traffic to over a specific period of time. The concept of the TDC has been borrowed from the references [71, 72]. The TDC shows traffic diversity. Various values of TDC are tuned to evaluate performance, both for workloads that exhibit low communication degree (low diversity), as well as for workloads that exhibit a much higher communication degree (high diversity). For a given value of TDC, the servers attached to a ToR switch send TDC simultaneous flows to servers attached to a remote ToR switch. Degree of ToR is set to 20 to provide an over-subscribed network with a ratio of 2 : 1 (i.e. 400 $Gbps$ total capacity for intra-rack versus 200 $Gbps$ total capacity for inter-rack) because it has been reported in studies [121, 122] that majority of data centre traffic remains within the source rack. Due to this over-subscription, half of the servers in a rack generate traffic which is destined to the intra-rack servers while the

rest generate traffic which is destined to the inter-rack servers.

3.5.3 Simulation Scenarios

The performance of the proposed technique is evaluated with both fast switch and slow switch approaches. The $FS = 1$ approach gives the best performance due to its fast switching. The performance of the proposed hybrid architecture is investigated and compared with these two best and worst case solutions.

The simulation models use a value of 10 *ms* for the switching time of the slow switch [70]. A value of 1 μs for the switching time of the fast optical switch is used because this is a conservative choice, although in some types of fast optical switches, this value can be as low as a few nanoseconds [27–29]. Each new request is delayed by the RTT of the control packet. The RTT includes the processing time of the control packet at the controller (T_{proc}) and the overhead (T_{oh}). The overhead time comprises the propagation delay (5 ns for 1 m optical fibre), the processing delay of control packets at the electronic switch and the O-E-O conversion delay. The aggregate value of T_{oh} is conservatively set to 1 μs, although all these delays are negligible (at most a few nanoseconds [70]). A high value of 50 μs for T_{proc} is conservatively selected, as software based controllers are efficient enough to process millions of requests per second with an average response time of less than 20 μs [130] while hardware based controllers can process a packet within a few nanoseconds [27]. The simulation models use values of 1 *ms* and 2 *ms* for burst aggregation to evaluate the impact of burst aggregation time on the performance of the system. A mean value of 10 *ms* is used for the ON period and a mean value of 2 *ms* for the OFF period, both of which are exponentially distributed. The simulation models use values of $1, 4$ and 8 for the TDC to evaluate the impact of traffic diversity on the performance of the system. The simulation time was set to 2 seconds.

3.6 Results and Discussion

The performance of the proposed HOSA network is evaluated by analysing average end-to-end delays. It is assumed that traffic within the same rack has negligible latency because ToR switches have the capacity to switch packets within nanoseconds. The latency of the inter-rack traffic is investigated so that the performance of the optical interconnect can be measured. The end-to-end delay is defined as the time between when a packet is generated by the source server and when the packet is received by the destination server.

The end-to-end delay is the sum of the packet delay incurred at the ToR switch (T_{ToR}), the packet transmission delay ($\frac{L_{packet}}{B_{core}}$) and the propagation delay (T_{prop}) from the source to the destination servers, and is given by the following equation.

$$Delay = T_{ToR} + \frac{L_{packet}}{B_{core}} + T_{prop} \tag{3.47}$$

where L_{packet} is the length of the packet in bits and B_{core} is the data rate from the ToR switch to the optical switch. The T_{ToR} is the sum of the packet queuing delay at the NIC (T_{queue}), the packet processing delay (T_{pr}), the packet delay for burst assembly ($T_{assembly}$), the packet delay until burst departure (T_{depart}) and the delay due to O-E-O conversion (T_{oeo}) and is given by the following equation.

$$T_{ToR} = T_{queue} + T_{pr} + T_{assembly} + T_{depart} + T_{oeo} \tag{3.48}$$

There is no queuing or processing delay at the optical switch due to the use of all optical switching.

Figures 3.14(a), 3.14(b) and 3.14(c) depict the results obtained with TDC values 1,4 and 8. Two values of burst aggregation timeout (1 *ms* and 2 *ms*) are used. Each plot shows end-to-end packet delay measured in milliseconds as a function of the offered load. The six curves represent different scenarios such as slow switch only ($FS = 0$), fast switch only ($FS = 1$) and proposed hybrid fast and slow switches ($FS = 0.5$) with two different timeout parameters.

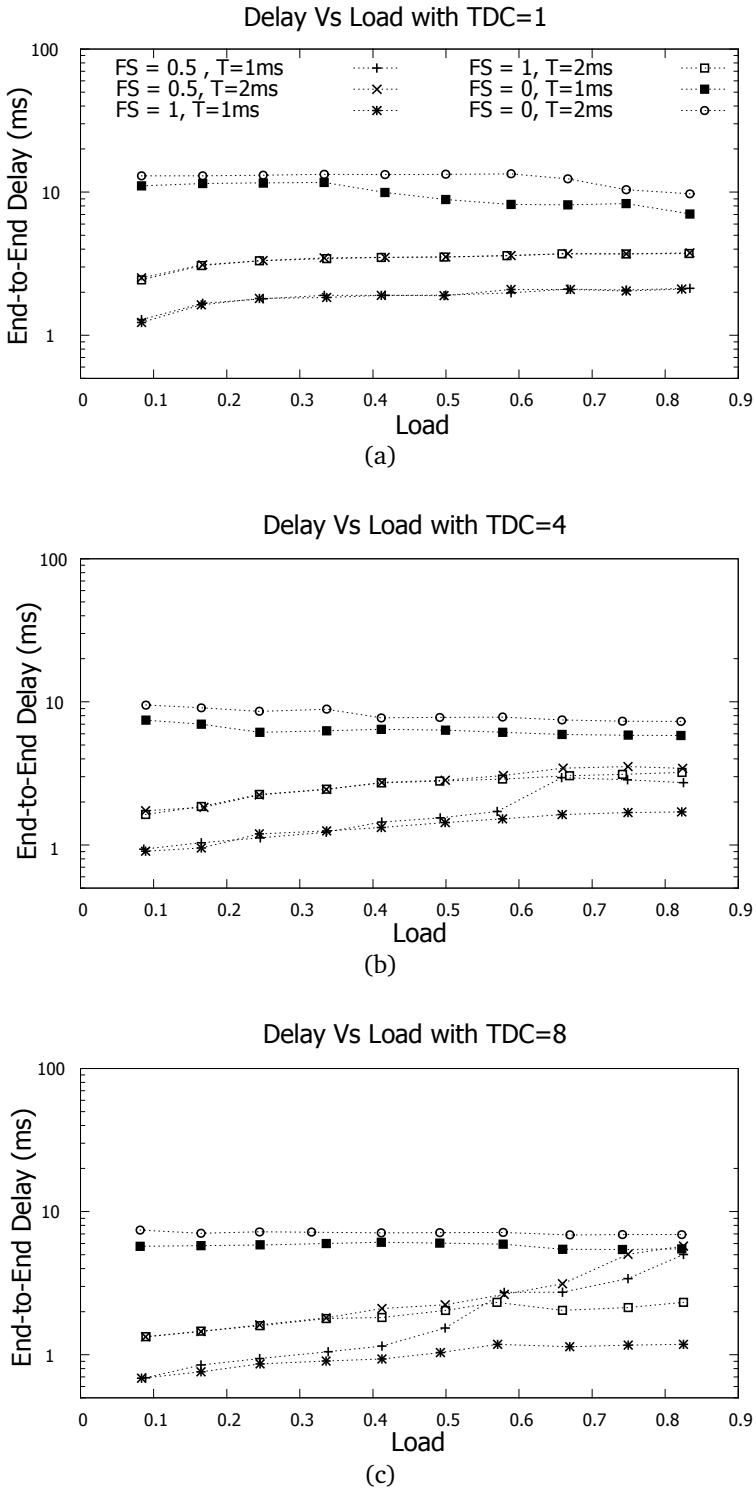
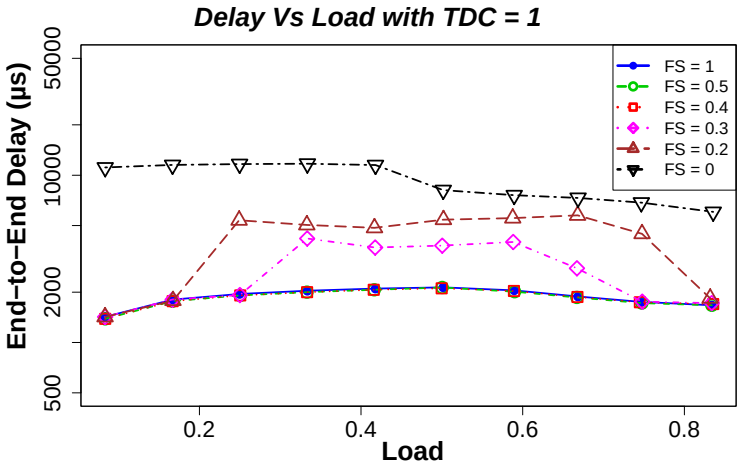
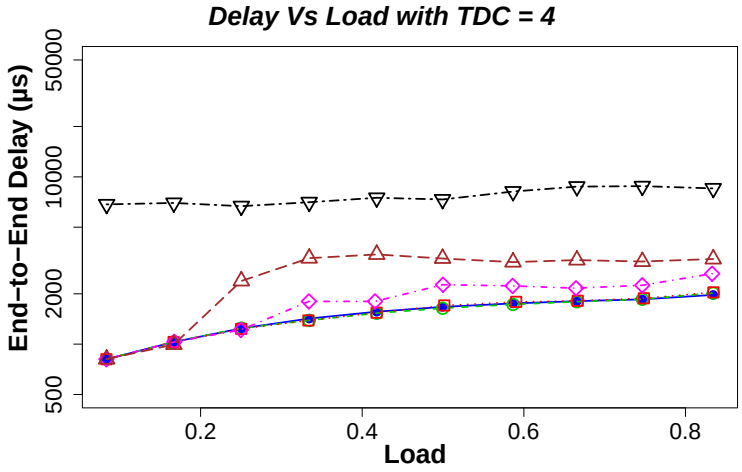


Figure 3.14. Load Vs End-to-End Delay for various timeout parameter and for various values of TDC: (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

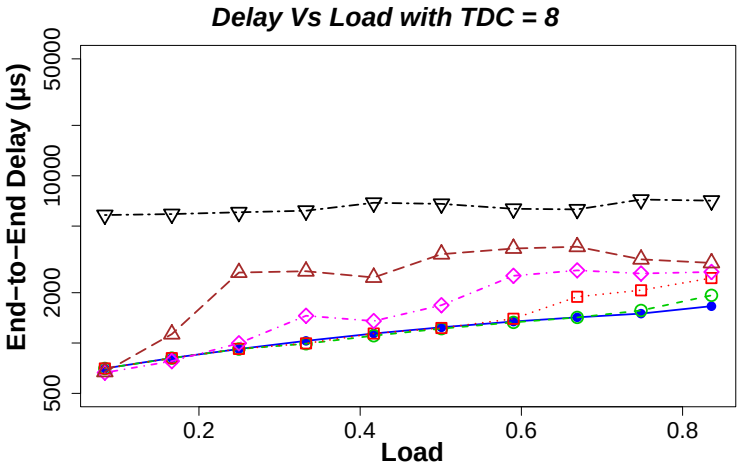
It can be seen in Figure 3.14(a) that with $TDC = 1$, the proposed hybrid technique with only half of its capacity provided by fast switches shows similar performance to



(a)



(b)



(c)

Figure 3.15. Load Vs End-to-End Delay for various capacities of fast and slow switches and for various TDC values: (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

that of a system using only fast switch till very high load i.e. 83% load. This is because with $TDC = 1$, aggregated traffic is high enough that can bypass the switching time of slow optical switch. So the fast optical switch is used during the reconfiguration phase of the slow optical switch. As a result, we get overall switching time that is comparable to the fast optical switch. Network congestion will be observed if we go beyond this 83% input load. This is because network bandwidth is wasted during the reconfiguration phase of slow optical switches and this will result in network congestion at very high load.

Figures 3.14(b) and 3.14(c) show that by increasing the TDC values, the performance of proposed interconnect technique with 1 *ms* timeout parameter at more than 50% load starts decreasing a bit with respect to the fast switch only approach but it is still better than the slow switch only approach. This is because with a high value of TDC, the duration of the generated burst is less than the switching time of the slow optical switch. More frequent requests of small size bursts are generated with high TDC values. So in the case of high TDC, traffic at loads less than 50% is routed through the fast switch paths since it uses half of the capacity of fast and slow switch but at high load, the congestion in the fast switch paths occurs which results in a high latency. The performance is improved slightly with high TDC values if we increase the timeout parameter which results in more traffic aggregation to generate large size bursts, but large timeout parameter also increases latency due to aggregation delay.

The performance of the proposed network degrades with the increase of TDC at high load. In order to overcome this limitation, the routing and scheduling algorithm with the speculation approach is introduced in section 3.2.5. The performance of this algorithm is investigated by implementing different capacities of fast switches (FS) as shown in Figure 3.15. In this analysis, a value of 1 *ms* is used for burst aggregation time. Figure shows that the proposed scheme with only 40% capacity of the FS demonstrates performance that is comparable to the best case with 100% FS till 83% load while the cost of the interconnect is reduced almost by one third as depicted in Figure 3.12(a).

3.6.1 Limitations

As discussed above, the proposed technique works well in the scenario when there is a large amount of traffic that can be divided among fast and slow switch paths. The performance is degraded with the increase in traffic diversity at high load. This problem can be avoided to a certain extent by increasing the traffic aggregation time for high diversity traffic but increasing the traffic aggregation time also increases latency. This limitation is addressed in the next chapter.

3.7 Conclusions

In this chapter, a novel optical interconnect called HOSA for data centre network is introduced. The hybrid architecture features MEMS OXCs for low cost and fast optical switches to achieve low latency. The core idea is to use fast optical switches to overcome the lengthy reconfiguration procedure of slow MEMS switches. The proposed design employs a single-stage core topology with multiple optical switches that has the capacity to be scaled up and scaled out easily.

OBS with a two-way reservation protocol is used to ensure zero burst loss. The two-way reservation is not suitable for long-haul backbone optical networks due to the high RTT of the control packet but for the proposed optical interconnect for the DCN, this RTT is not high.

A scalability analysis of the proposed interconnect, investigating various ratios of slow and fast optical switches, is presented. The proposed design is scalable to more than hundred thousand servers which is suitable for data centres of very large scale. In this chapter, a trade-off between the cost and power consumption of the proposed design by comparing it with conventional interconnects using analytical modelling is also presented. The results demonstrate its power efficiency as compared to other conventional interconnects. Almost 50% improvement in power consumption is achieved over Fat tree, BCube and traditional electrical network while $30-35\%$ improvement in power consumption is achieved over hybrid optical/electrical network.

Simulation is used to validate the proposed design. The trade-off between the performance and the capacity of both switches is evaluated. The results indicate that the proposed hybrid technique where only 40% of the interconnect capacity is provided by fast switches shows performance that is comparable to that of an interconnect exclusively using fast switches till 83% load, while the cost of the hybrid architecture is reduced almost to 33% compared to using fast switches only.

CHAPTER 4

HOSA WITH TRAFFIC DEMAND SCHEDULING

4.1 Introduction

In this chapter, improvements in the proposed design HOSA are introduced. The proposed changes relate to the control plane while there is no change in the functionality of the data plane. The large burst aggregation time and performance degradation in the case of high diversity traffic are the major limitations of the original design. These limitations are addressed in this chapter.

The target in this chapter is to reduce HOSA's large burst aggregation time to microseconds decreasing the overall latency. For this purpose, the concept of traffic demand scheduling (TDS) is proposed. In this technique, there is no need to aggregate large amounts of traffic. The controller maintains a traffic demand matrix which updates traffic demand periodically for each ToR pair and assigns slow paths to the ToR pairs that send high volume of traffic over a certain period of time. A resource allocation algorithm in the control plane is proposed for efficient utilization of the resources

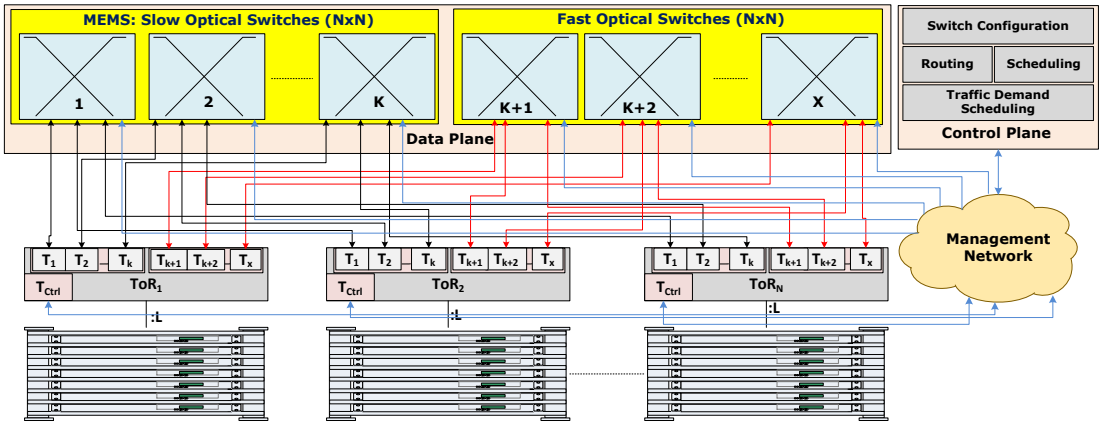


Figure 4.1. HOSA with Traffic Demand Scheduling

that results in high throughput and low latency. An unchanged technique of OBS with two-way reservation is used. Network-level simulation is used to evaluate the performance of the system using diverse workload communication patterns and system design parameters.

4.2 HOSA with TDS

The new design of HOSA with TDS for DCNs is shown in Figure 4.1. HOSA with TDS as before uses a two layer topology comprising electrical ToR switches at the edge and array of fast and slow optical switches at the core. Servers in a rack are connected with ToR switches using bidirectional fibre links. Each ToR switch has X optical transceivers, of which K transceivers are linked to the slow optical switches and $X-K$ transceivers are interfaced with the fast optical switches, where $1 < K < X$.

HOSA with traffic demand scheduling also features separate data and control planes. The control plane is realized by using a centralized controller. Routing, scheduling, switch configuration and traffic demand scheduling are the main tasks of the controller. Traffic demand scheduling is the new task of the controller that is introduced in this chapter. The controller collects traffic statistics to perform traffic demand scheduling. The traffic demand scheduling is used to configure the slow optical switches and

it ensures that elephant flows are routed through slow optical switches. The controller handles connection requests from all ToR switches, finds routes to the destination ToR switch through optical switches, assigns timeslots to the connection requests by selecting a suitable channel to the destination ToR switch, and configures optical switches with respect to the timeslots allocated. In order to realize these functions, the controller maintains a record of the global connectivity state of the optical switches. There is no change in the data plane. It is realized by using optical switches, performing data forwarding on pre-established lightpaths configured by the controller. Each ToR switch has a dedicated optical transceiver which is connected with the controller through a management network.

4.2.1 ToR Switch Design

There is no change in the design of the ToR switch and is shown in Figure 3.2. The ToR switch has an electronic switch fabric which is connected with the servers in the rack to perform intra-rack (within rack) switching in the electrical domain. To perform inter-rack (between racks) switching, it employs $(N-1)$ virtual output queues (VOQs) where N is the number of ToR switches in the network. There is a VOQ for each destination ToR switch in the DCN. Packets destined to the same ToR are aggregated into the same VOQ. Each VOQ is configured for a destination network address. The ToR switch maintains a VOQ table whose entries record the destination rack network address and the VOQ number. The dispatcher module matches the destination network address of the packet with the entry in this table and forwards the packet on the required VOQ.

4.2.2 Burst Assembly/Disassembly

Burst assembly can be timer based, length based or a combination of both approaches [39]. The original HOSA only considers a timer-based algorithm while in the HOSA with TDS, the hybrid approach is used in which either a timer expires or the burst length exceeds a threshold. The timer starts when a packet arrives at the empty VOQ. If the VOQ is not empty when the packet arrives, it joins the packets in the VOQ.

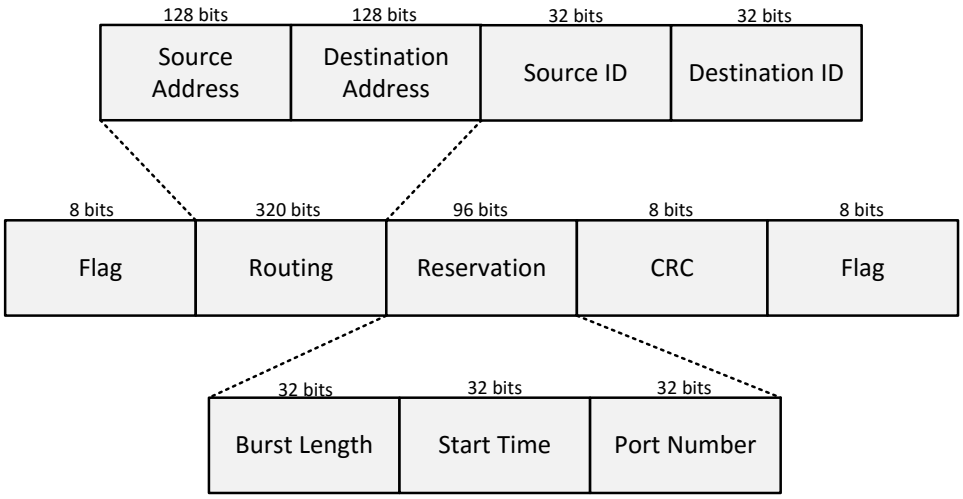


Figure 4.2. Control Packet Format for HOSA with TDS

The control packet is generated after the timer expires or the burst length exceeds the threshold and is sent to the controller using a transceiver dedicated to the control plane. The detailed description of the control packet and burst generation is presented in Chapter 5 in Algorithms 5 and 6.

The control packet that arrives at the controller contains fields recording the burst length, the IP addresses of the source and destination ToR switches and the IDs of the source and destination ToR switches. Each ToR switch is assigned a unique ID. The range of IDs of ToR switches is from 0 to $N-1$ in an N rack network. The controller processes the control packet, assigns the start time and the port number of the ToR switch on which a burst is to be transmitted and sends updated packet back to the source ToR switch. When the control packet arrives at the ToR switch, the scheduler module of the ToR switch generates a burst for the timeslot assigned by the controller. The timeslot refers to the duration of time assigned for a burst in an optical switch path. The generated burst is then sent to the queue of the allocated port. The scheduler module also initiates a new timer if the VOQ is not empty after the burst generation because new packets might have arrived during the RTT of the control packet. In order to realize bidirectional communication, the controller also generates a new control packet and sends it to the destination ToR switch. The destination TOR switch also generates a burst according to the timeslot allocated and sends it to the queue of the allocated port.

The ToR switch also has a burst disassembler and packet extractor module to disassemble the bursts received through the receivers. The receivers perform O-E conversion and sends the bursts to the disassembler module where packets are extracted from them and are sent to the electronic switch fabric and finally to the destination servers using electronic switching.

4.2.3 Control Packet Format

The new format of the control packet is shown in Figure 4.2. The control packet is 440 bits long and contains two main fields: routing and reservation. The routing field contains the source and destination IP addresses and the IDs of the ToR switches. These are the IP addresses of the network interface cards (NICs) reserved for the control plane in the ToR switches. The control packet has 128 bits for IPv6 addresses; however this length can be reduced to 32 bits when using IPv4 addresses resulting an overall control packet length of 31 bytes.

The reservation field is 96 bits long, and is divided into 3 sub-fields: 1) the burst length, 2) the start time and 3) the port number. In the original HOSA design, the reservation field required 4 additional subfields. The burst length field is filled by the ToR switch to request a timeslot from the controller. The controller fills the remaining two fields after processing the control packet. All of these three fields are 4 bytes long. The burst length field contains the burst length (expressed in bytes) while the start time contains the time (expressed in seconds) when the burst will be sent and the port number is the port of the ToR switch onto which the burst will be sent. The CRC field is reserved for cyclic redundancy check and a couple of optional fields are reserved for flags.

4.2.4 Control Plane Processing for HOSA with TDS

The controller performs routing, scheduling, traffic demand scheduling, and switch configuration functions. The routing operation deals with finding optimal slow and fast switch paths. The scheduling operation is related to schedule a burst on the fast

or on the slow switch path. The traffic demand scheduling is used to maintain traffic demand statistics for each ToR pair and it is also used to configure the slow optical switch so that elephant flows are routed via the slow path. The switch configuration is used to configure optical switches.

The routing and scheduling operation is performed when a control packet arrives at the controller for a new timeslot and is described in Algorithm 4. First, the controller extracts the source and the destination IDs of the ToR switches from the control packet. The next step is to check whether the same ToR pair has been assigned a timeslot recently to avoid duplicate timeslot allocation. The control packet is deleted if the difference of T_{cur} and T_{pre} is less than T_{dup}.

The next step is to find the latest horizon for both fast and slow paths. The term horizon refers to the latest available time when the channel will be free. The controller maintains a routing table which contains pre-defined routes of all source and destination ToR pairs and selects the best routes according to the latest horizon both for fast and slow switch paths (lines 8-11 in Algorithm 4). T_{fast} and T_{slow} represent the horizons of optimal fast and slow paths respectively. T_{est} represents the horizon of already established slow path between source and destination ToR pair and T_{RL} is the length of the burst expressed in time which is calculated on the basis of burst length (BL) in the control packet.

Table 4.1. Matrix Table

S|D	**0**			**1**			**2**		
	T	**CE**	**CA**	**T**	**CE**	**CA**	**T**	**CE**	**CA**
0	0	0	0	3531	1	2	1145	1	1
1	3531	1	2	0	0	0	1234	1	1
2	1145	1	1	1234	1	1	0	0	0

In order to configure the slow switch, the controller maintains a matrix table as shown in Table 4.1. It consists of three fields i.e. Traffic (T), Connections Exist (CE) and Connections Allowed (CA) for every source-destination ToR pair. The controller updates the traffic entry in the table for both source and destination ToR switches (lines

Algorithm 4 Control Plane Processing for HOSA with TDS

1: $cp \leftarrow controlpacket$
{**Above line shows that a control packet arrives at the controller and is assigned to** cp **object.** }
2: $srcID \leftarrow cp \rightarrow getSourceID()$
3: $destID \leftarrow cp \rightarrow getDestID()$
{**Above two lines extract source and destination IDs from the control packet and assign them to two variables (**$srcID$ **and** $destID$**).**}
4: $T_{pre} \leftarrow previousReservationTime(srcID, destID)$
5: **if** $(T_{cur} - T_{pre}) < T_{dup}$ **then**
6: $delete(cp)$
{**Condition in line 5 is used to avoid duplicate timeslot allocation. If true then the control packet is deleted because ToR pair has been assigned a timeslot recently.** }
7: **else**
8: $src\_fc \leftarrow findH\_Ch\_FS(srcID)$
9: $src\_sc \leftarrow findH\_Ch\_SS(srcID)$
10: $dest\_fc \leftarrow findH\_Ch\_FS(destID)$
11: $dest\_sc \leftarrow findH\_Ch\_SS(destID)$
{**Above four lines perform routing operation and return channels of latest horizon using minimum value search function. The routing operation results in finding total 4 channels in which there are 2 channels for source (1 for slow (**src_sc**) and 1 for fast path (**src_fc**)) and 2 channels for destination ToR (1 for slow (**$dest\_sc$**) and 1 for fast path (**$dest\_sc$**)).** }
12: $T_{fast} \leftarrow getMax(getH\_F(src\_fc), getH\_F(dest\_fc))$
13: $T_{slow} \leftarrow getMax(getH\_SS(src\_sc), getH\_SS(dest\_sc))$
{**In above two lines,** $getMax(value1, value2)$ **function is used to get the maximum value,** $getH\_F(channel)$ **function is used to get the horizon for fast path and** $getH\_SS(channel)$ **function is used to get horizon of slow switch paths. The maximum horizons for fast and slow switch paths are assigned to** T_{fast} **and** T_{slow} **respectively.**}
14: $src_{est} \leftarrow findH\_Ch\_SSest(srcID, destID)$
{**In above line,** $findH\_Ch\_SSest(srcID, destID)$ **function is used to find an already established slow switch path between ToR pair which is assigned to** src_{est} **variable.** }
15: **if** $src_{est} \neq NULL$ **then**
16: $T_{est} \leftarrow getH\_SS(est)$
17: $dest_{est} \leftarrow getDest\_ch(src_{est})$
18: **end if**
{**If condition in line 15 is false then it shows that an already established slow switch path does not exist. If condition is true then it finds the horizon of already established slow switch path using** $getH\_SS(channel)$ **function and assigns it to** T_{est}**. The** $getDest\_ch(channel)$ **method is used to get channel in destination ToR.**}

19: $BL \leftarrow cp \rightarrow getBurstLength()$
{**Above line extracts burst length from the control packet and assigns it to** BL **variable.** }
20: $T_{RL} \leftarrow BL * 8/Datarate$
{**Above line calculates burst length in time and assigns it to** T_{RL} **variable.**}
21: $src\_matrix \leftarrow getMatrixEntry(srcID, destID)$
22: $dest\_matrix \leftarrow getMatrixEntry(destID, srcID)$
{**Above two lines get matrix entries for traffic demand both for source and destination ToR switches.**}
23: $src\_matrix \rightarrow setT(src\_matrix \leftarrow getT() + (BL/1024))$
24: $dest\_matrix \rightarrow setT(dest\_matrix \leftarrow getT() + (BL/1024))$
{**Above two lines update matrix entries for traffic demand by adding requested burst length (expressed in kilo bytes) both for source and destination ToR.**}
25: **if** $(T_{cur} - T_{slow}) \geq T_{idle}$ **then**
26: $CE = matrix \rightarrow getNo\_of\_connectionsexist()$
27: $CA = matrix \rightarrow getMax\_connections\_allowed()$
{**Condition in line 25 is used to find a channel which is idle since** T_{idle}**. Lines 26 and 27 are used to get the number of connections already exist** (CE) **and the maximum number of connections allowed** (CA) **between the ToR pair respectively.**}
28: **if** $CE < CA$ **then**
29: $setupSlowPath(src\_sc, dest\_sc, T_{cur} + T_{sws} + T_{oh}, T_{cur} + T_{sws} + T_{guard}, T_{cur})$
{**Line 29 will establish a new slow path between ToR pair if conditions in line 25 and 28 are satisfied.**}
30: $src\_matrix -> setNo\_of\_connectionsexist(CE++)$
31: $dest\_matrix -> setNo\_of\_connectionsexist(CE++)$
{**Above two lines increment the number of connections exist both for source and destination ToR switches respectively.**}
32: **end if**
33: **end if**
34: **if** $(src_{est} = NULL \vee (T_{fast} + T_{RL}) < T_{est})$ **then**
{**A fast path is assigned to the current control packet if condition in line 34 is true.**}
35: $T_{start} \leftarrow T_{fast} + T_{swf} + T_{proc} + T_{oh}$
36: $T_{end} \leftarrow T_{start} + T_{RL} + T_{guard}$
{**Above two lines calculate and assign start and end times to** T_{start} **and** T_{end} **variables respectively.**}
37: $setupFastPath(src\_fc, dest\_fc, T_{start}, T_{end}, T_{cur})$
{**In above line,** $setupFastPath()$ **function is used to setup a new fast switch path.**}

38: $cp \rightarrow setStartTime(T_{start})$
{**Above line assigns the start time to the starttime field of the control packet.**}
39: $cpdest \leftarrow cp \rightarrow dup()$
{**Above line creates a duplicate control packet** ($cpdest$) **for bidirectional communication.**}
40: $cpdest \rightarrow setDestAdd(cp \rightarrow getSourceAdd())$
41: $cpdest \rightarrow setSourceAdd(cp \rightarrow getDestAdd())$
42: $cpdest \rightarrow setDestID(cp \rightarrow getSourceID())$
43: $cpdest \rightarrow setSourceID(cp \rightarrow getDestID())$
{**Above four lines set source and destination addresses and IDs in the duplicate control packet.**}
44: $cpdest \rightarrow setchannel(K + (dest\_fc \pmod{fast\_ch}))$
45: $cp \rightarrow setchannel(K + (src\_fc \pmod{fast\_ch}))$
{**Above two lines set the port number field both in the original and duplicate control packet.** }
46: $send(cp, T_{proc})$
47: $send(cpdest, T_{proc})$
{**Above two lines show that both control packets are sent after processing time.**}
48: **else**
49: $T_{start} \leftarrow T_{est} + T_{proc} + T_{oh}$
50: $T_{end} \leftarrow T_{start} + T_{RL} + T_{guard}$
{**Above two lines calculate and assign start and end times to** T_{start} **and** T_{end} **variables respectively.**}
51: $updateSlowPath(src\_est, dest\_est, T_{start}, T_{end}, T_{cur})$
{**In above line,** $updateSlowPath()$ **function is used to update already established slow switch path with a new horizon.**}
52: $cp \rightarrow setStartTime(T_{start})$
53: $cpdest \leftarrow cp \rightarrow dup()$
54: $cpdest \rightarrow setchannel(dest\_est \pmod{slow\_ch})$
55: $cpdest \rightarrow setDestAdd(cp \rightarrow getSourceAdd())$
56: $cpdest \rightarrow setSourceAdd(cp \rightarrow getDestAdd())$
57: $cpdest \rightarrow setDestID(cp \rightarrow getSourceID())$
58: $cpdest \rightarrow setSourceID(cp \rightarrow getDestID())$
59: $cp \rightarrow setchannel(src\_est \pmod{slow\_ch})$
60: $send(cp, T_{proc})$
61: $send(cpdest, T_{proc})$
{**Lines 52 to 61 are used to generate a duplicate control packet and to fill required fields in the original and duplicate control packet. In the end, both are sent after processing time.**}
62: **end if**
63: **end if**

23-24 of Algorithm 4). A new parameter i.e. Intervaltime ($T_{interval}$) is used by the controller to update the entry for CA in the matrix table. This parameter is initialized to a fix value of time in milliseconds when the controller starts its operations. The CA is the maximum number of connections for slow path that a given source destination ToR pair can have in $T_{interval}$ and is calculated by the following formula.

$$CA = \lceil \frac{T \times 1024 \times 8}{datarate \times T_{interval}} \rceil \tag{4.1}$$

T_{idle} represents the idle time for which the channel has not been used. The algorithm sets up a new slow path if $T_{cur} - T_{slow} \geq T_{idle}$ and $CE < CA$ as shown in lines 25-29 in Algorithm 4. It assumes that the channel in the slow switch can be assigned to the new request if it is idle since T_{idle} and the traffic matrix also allows this i.e. $CE < CA$. All new paths of the slow switch are assigned on the basis of this principle. However, the current control packet request is assigned to a fast path if there is no slow path already established or $(T_{fast} + T_{RL}) < T_{est}$). Otherwise, the already established slow path is assigned to the current request by updating its horizon in the controller. After assigning timeslots, the start time and port number fields in the control packet are filled by the controller (lines 38,45). The controller also generates a new control packet by duplicating the existing control packet and updates its port number, source and destination addresses and their relevant IDs because the source and destination ToR will have different port numbers (lines 39-44,53-58). Finally, the original control packet is sent back to the source ToR switch and the newly generated control packet is sent to the destination ToR switch for bidirectional communication (lines 46-47,60-61).

Figures 4.3, 4.4 and 4.5 represent a graphical representation of the scenarios for timeslot allocation. There are two parts in Figures 4.3 and 4.4, (a) represents the channel states before timeslot allocation i.e. when a control packet arrives at the controller and (b) represents the channel states after the timeslot has been allocated for the requested timeslot duration. Figures 4.3 and 4.4 have three channels for fast switch paths and three channels for slow switch paths while Figure 4.5 has only channels for slow switch paths because it illustrates the establishment of a new path only

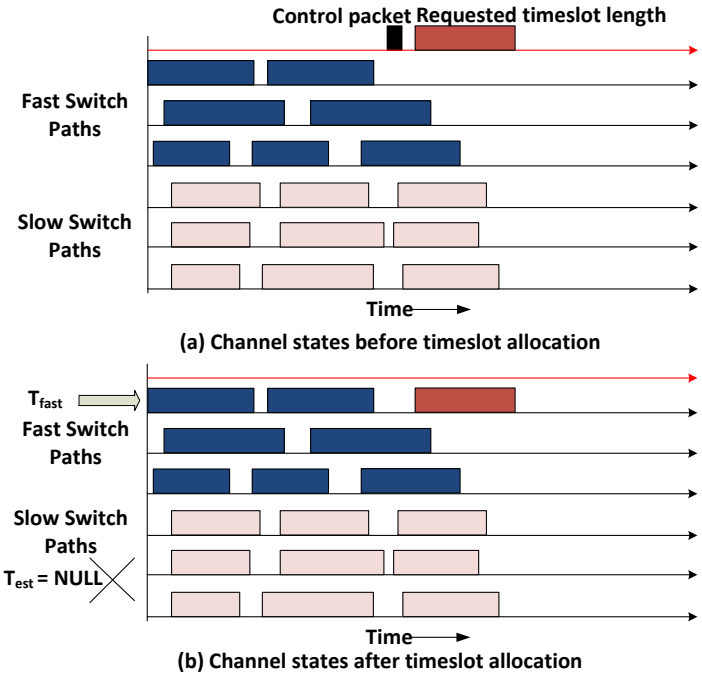


Figure 4.3. Timeslot Allocation for HOSA with TDS: Case 1

in the slow switches.

Figures 4.3 shows a scenario when an incoming burst is assigned a timeslot in a fast switch path i.e. the first channel in the fast switch paths. In this scenario, $T_{est} = NULL$ i.e. no already established connection exists. In HOSA, timeslots in some scenarios are divided between fast and slow switch paths, but in HOSA with TDS, timeslot is not divided and it is assigned only in the fast or in the slow switch path. Figures 4.4 presents a scenario when an already established slow switch path is assigned for an incoming burst because condition $(T_{fast} + T_{RL}) < T_{sch})$ is false.

Figures 4.5 illustrates a mechanism for the establishment of a new slow switch path. This path is established for future incoming requests, and the current request is assigned a path either in an already established slow or in a new fast path as shown in Figures 4.3 and 4.4. A new slow switch path will only be established if condition $T_{cur} - T_{slow} \geq T_{idle}$ and $CE < CA$ is satisfied.

All ports of the slow optical switches may not be connected at all the time. Fortunately, connectivity is easy to achieve via the port exchange operation as described in [54]. First, it finds all unconnected ports. It then selects two unconnected ports

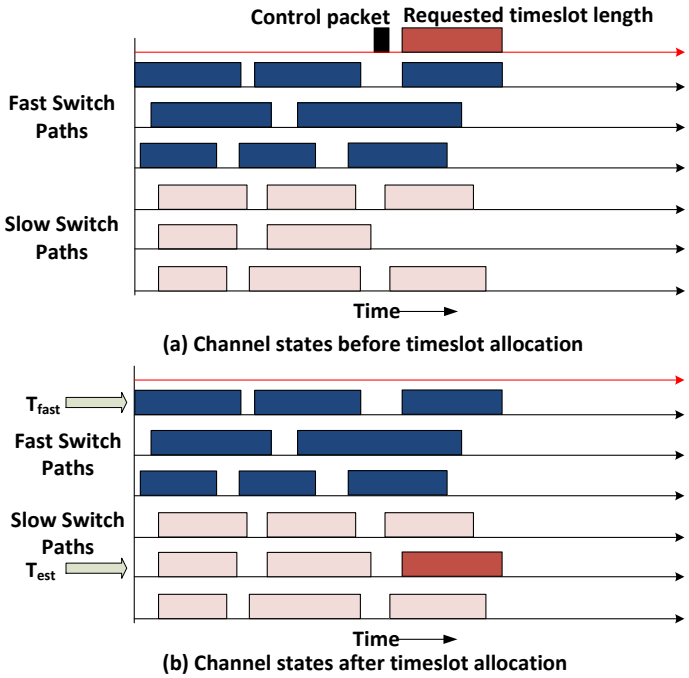


Figure 4.4. Timeslot Allocation for HOSA with TDS: Case 2

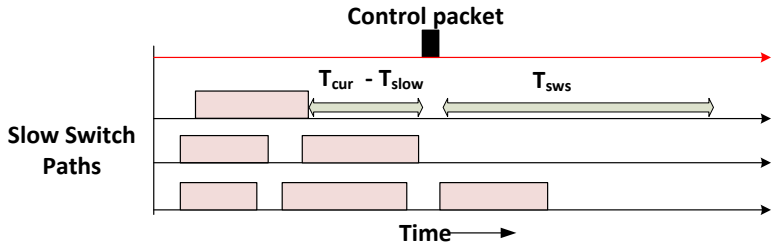


Figure 4.5. Establishment of a new slow switch path.

$a \rightarrow b$ from all unconnected ports and two connected ports $c \rightarrow d$ and connects them by replacing links $a \rightarrow b$ and $c \rightarrow d$ with $a \rightarrow c$ and $b \rightarrow d$. There is a daemon process in the controller which runs periodically to check and connect the unconnected ports in the slow optical switches.

Switch configuration is the last operation of the controller, and is performed as in the original HOSA design. After processing the control packet, a configuration message is generated and is sent to the switch controller to configure the optical switch.

4.3 Performance Analysis

To assess the latency and throughput performance of HOSA with TDS, the control logic of the simulations model developed for HOSA is changed appropriately. The models for ToR switches are also changed to support the mixed timer/length based approach to burst assembly.

The simulation topology consists of $T_{RK} = 40$ total racks. Each rack has $S_{RK} = 40$ servers and 1 ToR switch. Servers are connected to the ToR switch using bidirectional fibre links. Each ToR switch is also linked with the electrical switch using a bidirectional fibre link via a transceiver reserved for use by the control plane. The electrical switch in the control plane is connected with all the ToR switches, the controller and all the optical switches. The simulation topology considers two optical switches, one for the slow path and other for the fast path. It also considers 2:1 core over-subscription by using $X = 20$ optical transceivers per ToR switch connected to the optical switches. To evaluate performance at different switching capacities for slow and fast optical switches, $K = \{0, 10, 12, 14, 16\}$ links for the slow optical switch and $X - K = \{20, 10, 8, 6, 4\}$ links for the fast optical switch are used.

4.3.1 Traffic Generation

As described in the previous chapter, the traffic characteristics of data centres is bursty in nature and shows evidence of ON-OFF behaviour [121]. The simulation models use a Markov Chain Process model for bursty traffic with an ON period of 800 μs and an OFF period of 200 μs which are exponentially distributed. Various exponential inter-arrival rates of packets during the ON period are considered to investigate traffic at different loads. In this chapter, two parameters are defined to control traffic generation similar to other designs [71, 72]. These are:

- Stability: It is the lifetime (in milliseconds) of a traffic flow between two ToR switches.

Table 4.2. Simulation Parameters for HOSA with TDS

Parameter Name	Symbol	Value
Racks/ToR Switches	T_{RK}	40
Servers per rack	S_{RK}	40
Fast & Slow Switch		1 Each
Electrical Switch for control plane		1
Degree of ToR Switches	X	20
Degree of ToR to MEMS	(K)	{0,10,12,14,16}
Degree of ToR to Fast Switch	$(X-K)$	{20,10,8,6,4}
Control packet processing time	T_{proc}	$1\mu s$
Switching Time of Slow Switch	T_{sws}	10 ms
Switching Time of Fast Switch	T_{swf}	$1\mu s$
Intervaltime	$T_{interval}$	2 ms
Overhead	T_{oh}	$1\mu s$
Data rate		10 Gbps
ON Period	T_{ON}	$800\mu s$
OFF Period	T_{OFF}	$200\mu s$
Burst Aggregation Time	T_a	$100\mu s$
Maximum Burst Length	BL	$500KB$
Topological Degree of Communication	TDC	{1,10,20} racks
Stability	St	{100,200,300,400,500} ms

- Topological Degree of Communication TDC: This is the number of simultaneous destination ToR switches that a given source ToR switch sends traffic to.

The TDC parameter represents the diversity of traffic workloads. Different values of TDC parameters are used to evaluate performance at low, medium and high traffic diversity. The stability is used to evaluate the ability of the core interconnect to adapt to constantly changing communication patterns.

4.3.2 Simulation Parameters

The key simulation parameters are shown in Table 4.2. A value of 1 μs is used for the processing time of the control packet by the controller. The simulation models use a value of 10 *ms* for the switching time of the slow MEMS switch [54]. A value of 1 μs is used for the switching time of the fast optical switches; this is a conservative

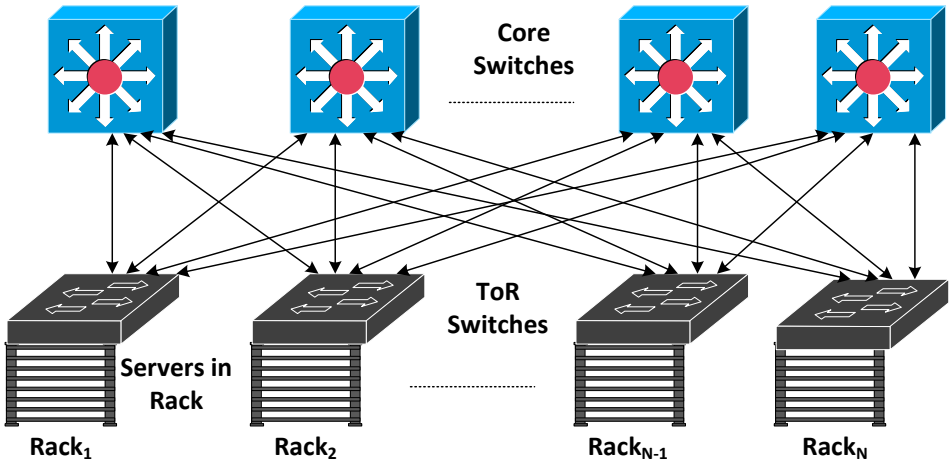


Figure 4.6. Topology diagram for the baseline traditional electrical network (Leaf-spine topology)

choice, although in some types of fast optical switch, this value can be as low as a few nanoseconds [27]. The RTT of the control packet includes its processing time at the controller and the overhead time. The overhead time comprises propagation delay (5ns for 1m optical fibre), the processing delay of the control packet at the electrical switch, and the O-E-O conversion delay. The aggregate value of overhead time is conservatively set to 1 μs although all these delays are negligible (at most a few nanoseconds [54]).

The simulation models use a value of 2 ms for Intervaltime ($T_{interval}$) for matrix calculation. For burst generation, a combination of 100 μs for aggregation time (T_a) and 500 KB for burst length (BL) is considered. Three cases for TDC are considered by using values drawn from the set $\{1, 10, 20\}$ and five cases are considered for stability by using values drawn from the set $\{100, 200, 300, 400, 500\}$ ms in order to evaluate their impact on performance of the system. Simulation time was set to 2 seconds.

4.3.3 Baseline Electrical Network

The performance is benchmarked against an ideal traditional electrical (TE) packet switching network that features a two layer leaf-spine topology [131] as shown in Figure 4.6. Its latency and throughput performance provides a baseline against which to compare the performance of the new networks. The TE network acts as an ideal

electrical packet switching network that has low latency and high throughput as compared to the FT, BCube and OE networks due to the higher number of hops in these networks.

4.4 Results and Discussion

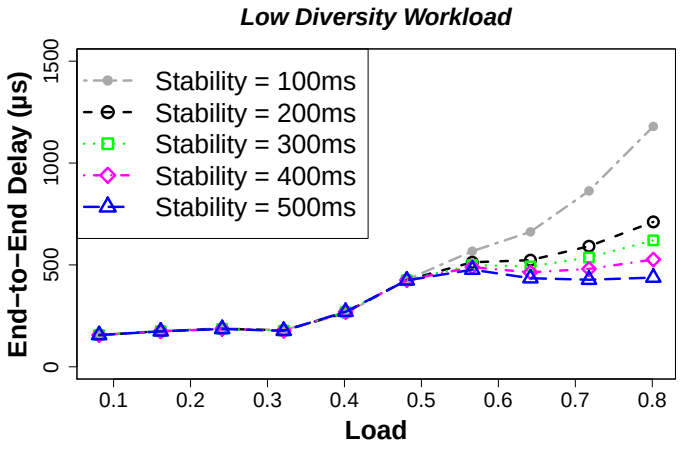
4.4.1 Latency

The term end-to-end delay is used to measure the latency. The latency of Intra-rack traffic is not measured because it is negligible due to fast switching of electrical switches, so only the latency of inter-rack traffic is measured so that the performance of optical interconnect can be evaluated.

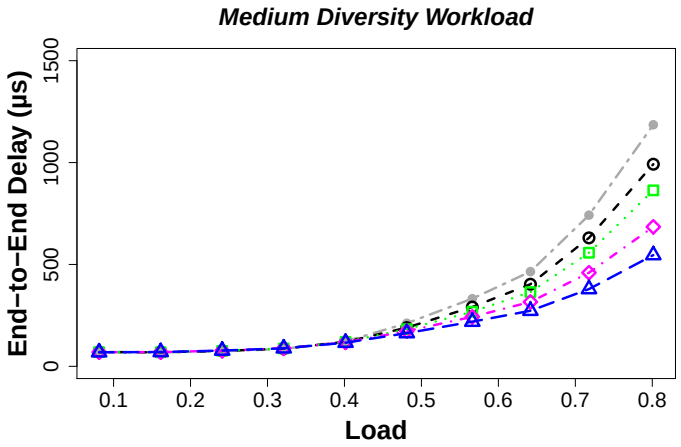
Multiple runs were executed using the traffic patterns and other simulation parameters described in Table 4.2. Figure 4.7 depicts the results obtained with $TDC = \{1, 10, 20\}$ respectively for various values of TDC and with various values of stability for equivalent capacities of fast and slow switches.

All three figures that the traffic stability has negligible impact on end-to-end latency of packets for loads below 50% while latency increases with decreasing stability value at higher loads. This is because 50% of the capacity of fast optical switches is enough to carry traffic at loads upto 50%. The stability parameter has an inverse relation with latency because the slow optical switches need to be reconfigured after stability period. A low stability value results in more frequent reconfigurations of MEMS switch which ultimately increases the latency due to the long switching time of MEMS at higher levels of load, while at lower load levels the traffic is carried through the fast path without having any impact of stability on the latency.

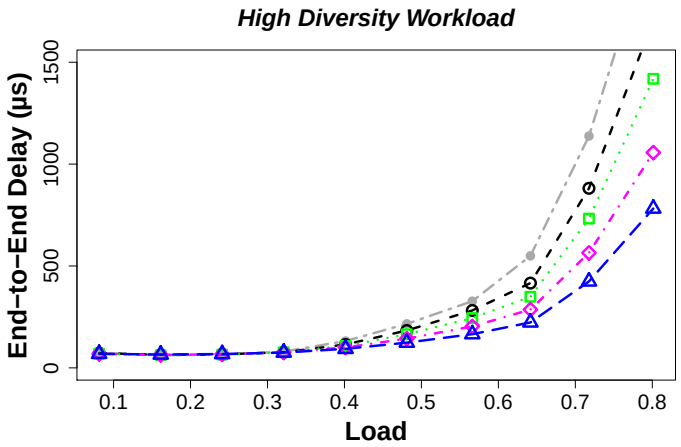
TDC also has a direct impact on the latency at higher load. Figures 5.4(a), 5.4(b) and 5.4(c) show that latency with TDC=20 almost doubles with respect to the latency with $TDC = 1$ at higher load. This is because with a higher TDC value at higher load, more control packet requests are generated which consume more timeslots. As the number of timeslots increases, more time is wasted as overhead, processing and



(a)



(b)



(c)

Figure 4.7. Load Vs End-to-End Delay with various values of stability parameter for various TDC values using equivalent capacities of fast and slow optical switches. (a) TDC = 1, (b) TDC = 10, (c) TDC = 20.

switching time. For example with $TDC = 1$, five control packets are generated at higher load by each ToR switch in a 100 μs duration by the hybrid scheduling for burst assembly while with $TDC = 20$, twenty control packets are generated.

Simulation results showing latency for various capacities of fast and slow optical switches and for the baseline electrical network are shown in Figure 4.8. Figure 4.8 shows delay versus offered load for three values of TDC at high stability (i.e. at 500 ms). The results for the baseline electrical network act as the performance benchmark while the results for $FS = 1$ act as the best case in which all of the switching capacity is provided by the fast switches only. The improvements of the proposed design in scalability, cost and power consumption can be seen comes to at the cost of increased latency. This is due to the traffic aggregation delay that is an inherent limitation of optical burst switching and to the higher switching time of MEMS switches, but the performance is comparable to that of the baseline electrical network.

4.4.2 Throughput

The observed throughput of each link from the ToR switch to the optical switch is given by using the following formula.

$$Th_{perlink} = \frac{Total_{bits}}{T_{sim}} \tag{4.2}$$

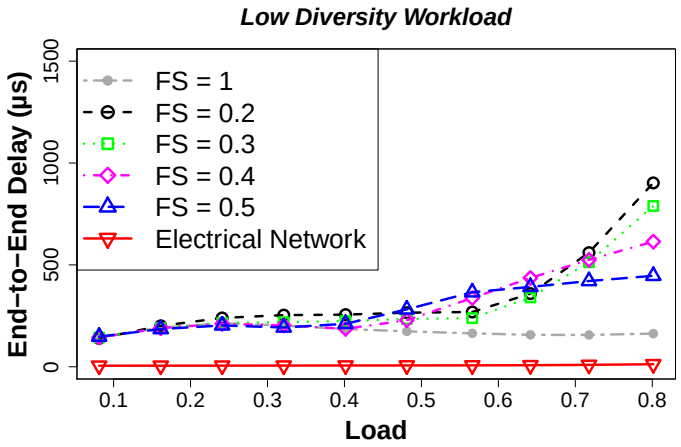
where $Total_{bits}$ is the total number of bits successfully delivered through the link in the optical switch and T_{sim} is the total simulation time. The average network throughput per link is calculated as:

$$Th_{avg} = \frac{\sum_{n=1}^{N_{link}} Th_{perlink}[n]}{N_{link}} \tag{4.3}$$

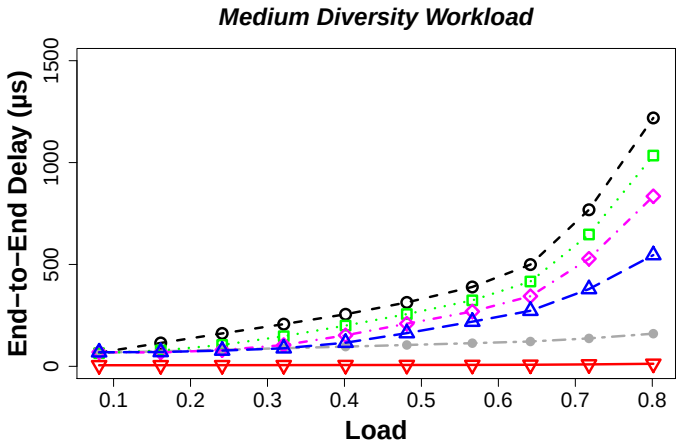
where N_{link} is the total number of links of ToR switches connected with optical switches.

Figure 4.9 shows the throughput observed at 80% offered load at the core for three values of TDC at high stability.

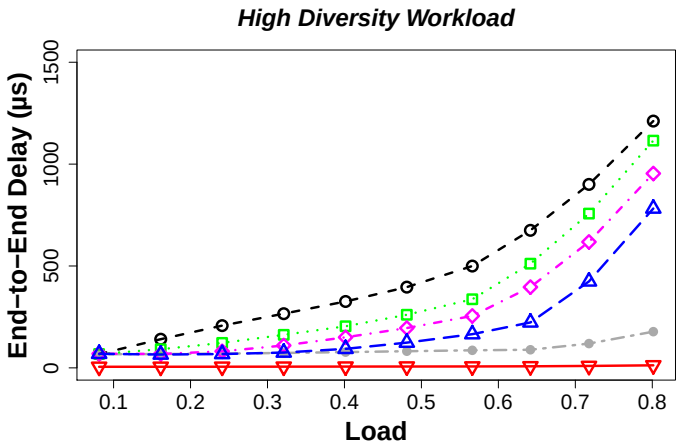
Figure 4.9 shows that the average bandwidth of slow optical switches is higher than that of the fast optical switches. This is because the majority of the traffic is



(a)



(b)



(c)

Figure 4.8. Load Vs End-to-End Delay for various capacities of fast and slow switches with various TDC values, at high stability: (a) TDC = 1, (b) TDC = 10, (c) TDC = 20.

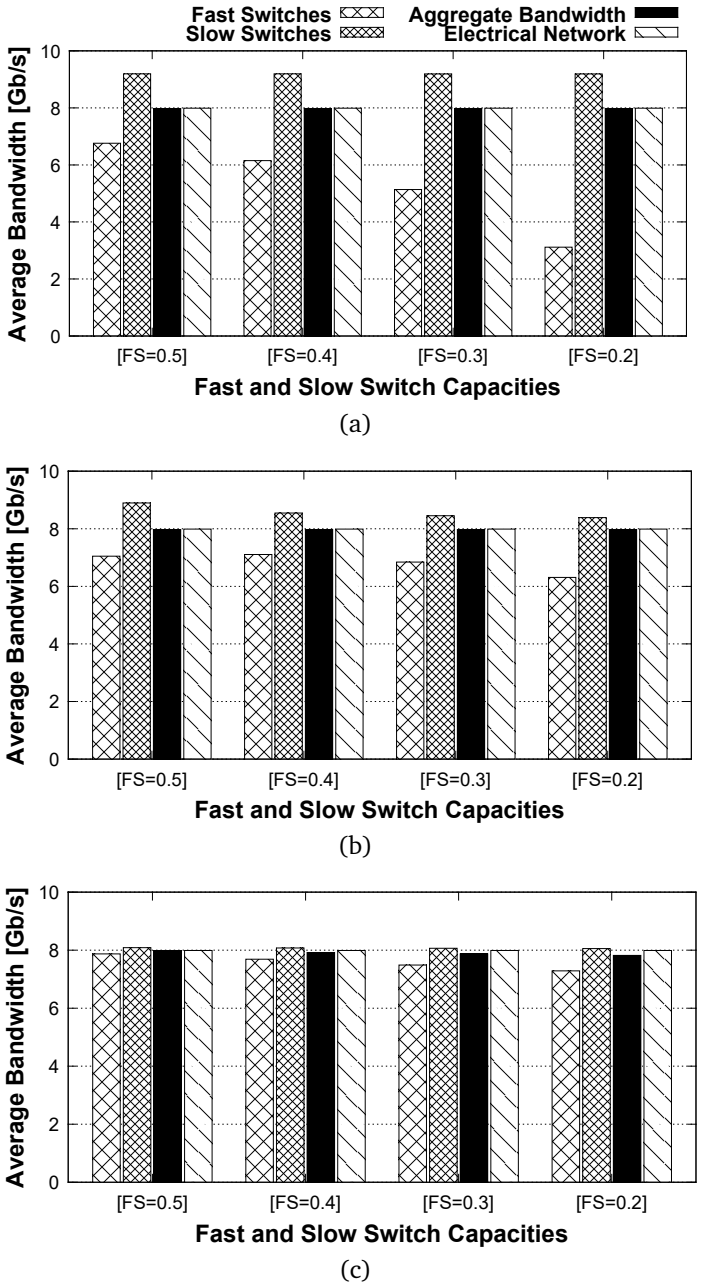


Figure 4.9. Average Bandwidth in (Gb/s) for various capacities of fast and slow switches with various TDC values: (a) TDC = 1, (b) TDC = 10, (c) TDC = 20.

routed through slow optical switches. This decreases the overall power consumption of the interconnection network because slow MEMS switches are more power efficient than to fast optical switches due to featuring passive switching.

It can also be seen from Figure 4.9(a) and 4.9(b) that the overall interconnect bandwidth using both type of switches remains close to 8 $Gbps$ at 80% load with

$TDC = 1$ and $TDC = 10$. It decreases slightly when $TDC = 20$ as shown in Figure 4.9(c) but is still comparable to the bandwidth of the baseline electrical network. This is because with high diversity traffic, there are plenty of requests for new connections and each request is delayed by the RTT of the control packet. The bandwidth of the interconnection is wasted during this RTT which results in decreasing overall network throughput in the presence of high diversity traffic. This behaviour is also observed in other optical interconnects [54, 72]

4.5 Performance of the Control Plane

In order to assess the performance of the routing and scheduling algorithm of the control plane, the algorithm was run on an Intel host with a Core i7, 2.17 GHz processor and 16 GB RAM. The results were obtained for several combinations of parameters. For statistical significance, the results of 1000 runs were averaged and the results are shown in Table 4.3. Table 4.3 shows the execution time for various numbers of racks N, for various values of topological degree of communication TDC, and various number of slow and fast optical switches. Although these execution times are implementation dependent, their variations illustrate the scalability properties of the algorithms in the control plane.

When a control packet arrives at the controller, the controller implements the routing and scheduling operation of Algorithm 4. The complexity of the routing and scheduling algorithm is $O(2(2K + L) + \mu)$, where K is the number of ports of the ToR switch dedicated for the slow switch paths, L is the number of ports of the ToR switch assigned for the fast switch paths and μ represents the aggregate processing time of all other instructions. This is assumed to be a constant of negligibly low value. We measure the algorithm execution time in a 4:1 oversubscribed network when $\{K, L\} = 5$, in a 2:1 oversubscribed network when $\{K, L\} = 10$ and in a fully subscribed network when $\{K, L\} = 20$ using 40 servers per rack as shown in first three rows of Table 5.3. Fourth row of Table 5.3 shows its execution in a fully subscribed networking using 80 servers per rack. It can be inferred that the processing time of the control packet

Table 4.3. Performance of the Algorithms in the Control Plane

Algorithm	*Racks (N)*	*TDC*	*K*	*L*	*Exec.T*
Routing and scheduling	$\forall$ N	$\forall$ TDC	5	5	< 0.1 μs
			10	10	< 0.1 μs
			20	20	< 0.5 μs
			40	40	1.1 μs
Traffic Matrix Scheduling	102	1	$\forall$ K	$\forall$ L	5.6 μs
		10			23.6 μs
		20			42.9 μs
		101			204.9 μs
	512	1			27.7 μs
		10			116 μs
		20			211.9 μs
		511			7.3 ms
	1024	1			51.8 μs
		10			229.9 μs
		20			426.9 μs
		50			1.1 ms
		100			2.1 ms
		1023			29.2 ms

is independent of the network size and the TDC values. The execution time of the routing and scheduling algorithm is very low in 4:1 and 2:1 oversubscribed networks while it increases slightly because of the increased number of ports of ToR switches in a fully subscribed network.

The traffic demand scheduling is used to measure traffic statistics to configure the slow optical switches. The complexity of this algorithm is $O(N \times (N-1) + \mu)$. The performance of this algorithm depends upon the network size and TDC parameter. It is independent of the network over-subscription as shown in Table 5.3. This algorithm runs periodically to predict the new traffic matrix. It can be seen that the execution time is proportional to network size and TDC. In a very large network in a worst case

scenario, with $N = 1024$ and $TDC = 1023$, the execution time around 30 *ms* was observed that is understandably high, but in a real network scenario, the TDC would not be too high because different studies on data centre traffic [102, 119, 121] have shown that traffic within data centres is bounded in degrees and racks communicate with only few other racks over a given period of time.

4.6 Conclusion

In this chapter, the performance of the original design HOSA is improved using traffic demand scheduling. In this technique, the controller maintains a traffic demand matrix which updates traffic demand periodically for each ToR pair and assigns slow paths to the ToR pairs that send high volume of traffic over a certain period of time. A resource allocation algorithm in the controller is proposed that ensures minimum latency and high throughput.

The network-level simulation investigating various traffic scenarios for stability and workload diversity, and considering various capacities of slow and fast optical switches is used to validate the proposed design. The results that the performance of HOSA has been improved by introducing HOSA with TDS scheme. Low latency and high throughput has been achieved with various workload communication patterns and that performance is comparable to that of electrical data centre networks for low and medium traffic loads.

In the next chapter, a new design for DCNs is proposed that is based on faster switching technologies that are now available [28, 29, 59, 64]. Instead of using OPS with fast optical switches, OBS with two-way reservation protocol is considered.

CHAPTER 5

PERFORMANCE ANALYSIS OF OBS OVER FAST OPTICAL SWITCH ARCHITECTURE FOR DCN

5.1 Introduction

The performance of optical network is directly related to the type of the optical switching technique used. These switching techniques are OCS, OPS and OBS. The MEMS OXC or OCS switch has been used in the backbone optical network for many years. Hybrid designs for data centre networks that use OCS in conjunction with other technologies have been proposed [23,50–54]. Helios and cThrough [23,50] propose using OCS in conjunction with traditional electrical packet switching while the LIGHTNESS project [51,52] employs OCS together with optical packet switching. The Hydra, OSA and Reconfigurable designs [53,54,72] augment OCS with a multi-hopping technique. A major issue with these interconnects has been their slow reconfiguration time due to the limitation of 3D-MEMS technology [26]. This reconfiguration time is influenced by

two factors: (1) the switching time of the 3D-MEMS switch i.e. 10-100 ms [23,50,54], and (2) the software/control plane overhead required for the estimation of traffic demand and for the calculation of a new OCS topology i.e. 100 ms to 1 s [31]. Consequently, the control plane can only support applications that have high traffic stability, i.e. workloads that last several seconds [23]. This limitation of control plane also exists in the proposed hybrid architecture HOSA with TDS. To address this limitation, a new design of optical interconnect for data centre is proposed in this chapter.

The new proposed design for DCNs called as FOSA makes use of the faster switching technologies that are now available [28, 29, 59, 64]. These switches are described in Chapter 2. Instead of using OPS with fast optical switches, again OBS with two-way reservation protocol is considered. Various diverse workloads traffic patterns with various data rates in networks having different edge to core oversubscription ratios are considered. The performance of such designs across usage patterns is evaluated. The more detailed description of the FOSA is provided in the next section.

5.2 Fast Optical Switch Architecture: FOSA

The new design FOSA also considers OBS. Packets are aggregated to produce bursts of short duration. A control packet is created to request the allocation of resources needed to transmit the burst from the controller by using a two-way reservation process. The controller assigns resources and sends the control packet back to the originating node as an acknowledgement. The burst is then transmitted on the pre-established path configured by the controller. Unlike HOSA with TDS, the controller in the FOSA does not keep track of traffic demands between ToR pairs. This reduces any additional burden on the control plane in the FOSA.

The FOSA for DCNs is illustrated in Figure 5.1. The proposed topology has two layers, i.e., edge and core. The edge contains electronic ToR switches while the core comprises a group of fast optical switches. Servers in each rack are connected to the ToR switches using bidirectional optical fibres. The ToR switches are linked to the optical switches using unidirectional optical fibres.

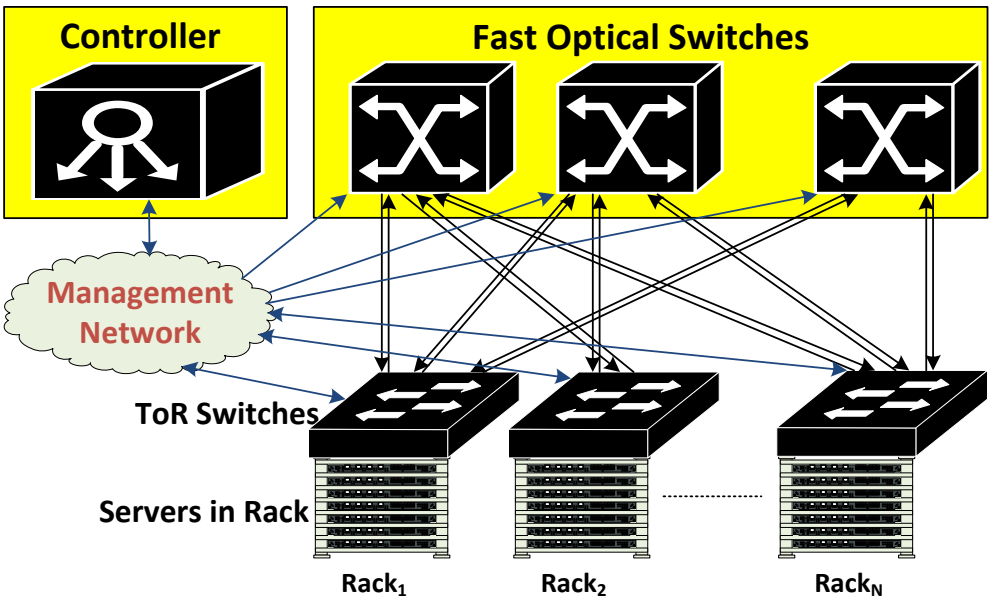


Figure 5.1. Fast Optical Switch Architecture.

Similar to the previous designs, the FOSA also features separate control and data planes. The control plane comprises a centralized controller. The controller performs routing, scheduling and switch configuration functions. It receives connection setup requests from all ToR switches, finds routes, assigns timeslots to the connection requests, and configures optical switches with respect to the timeslots allocated. The data plane comprises optical switches that perform data forwarding on pre-configured lightpaths set up by the controller. A management network is used by the control plane that connects every ToR switch to the controller via a transceiver in each ToR switch reserved for use by the control plane.

There is no change in the design and functionality of the ToR switch and the control packet format from Chapter 4. For burst assembly, the mixed approach is used in which either a timer expires or the burst length exceeds a threshold. The procedure for traffic aggregation is described in Algorithm 5. The timer starts when a packet arrives at the empty VOQ in the ToR switch. If the VOQ is not empty when the packet arrives, it joins other packets in the VOQ (lines 11-16 in Algorithm 5). The control packet is generated after the timer expires or the burst length exceeds the threshold

and is sent to the controller using transceiver dedicated for the control plane (lines 1-8 in Algorithm 5). The control packet at this stage contains information of the burst length, IP addresses of source and destination ToR switches, and IDs of the source and the destination ToR switches. Each ToR switch is assigned a unique ID. The range of IDs of ToR switches is from 0 to $N-1$ in an N rack network. These IDs are used by the controller to perform routing and scheduling algorithm. The controller processes the control packet, assigns a start time and a port number of the ToR switch on which the burst is to be transmitted and sends it back to the source ToR switch.

When the control packet arrives back at the ToR switch, the scheduler module of the ToR switch generates a burst according to the timeslot assigned by the controller. The timeslot refers to the duration of time assigned for a burst in an optical switch path. The generated burst is then sent to the queue of the allocated port. The scheduler module also initiates a new timer if the VOQ is not empty after the burst generation because new packets might have arrived during the RTT of the control packet. The process of burst transmission is explained in Algorithm 6. In the first step, the information of burst size, port number and start time is extracted from the control packet that arrives at the ToR switch after being processed by the controller (lines 1-4 in Algorithm 6). Then a burst is generated and packets are extracted from VOQ and added into burst (lines 5-14 in Algorithm 6). The burst is then transmitted on the assigned port number at the assigned timeslot (line 15 in Algorithm 6). After transmitting the burst, the parameters are reinitialized (lines 16-23 in Algorithm 6).

The ToR switch also has burst disassembler and packet extractor module to disassemble the bursts received through the receivers. The receivers perform optical to electrical conversion and send bursts to the disassembler module where packets are extracted from them and are sent to the electronic switch fabric and finally to the destination servers using electronic switching.

Burst assembly cycle is shown in Figure 5.2. Packets are aggregated to make a burst. Burst assembly time is represented by T_a. The control packet is sent by the ToR switch to the controller for resource reservation by using a management network. The time taken by a control packet to reach the controller is called overhead time

Algorithm 5 Traffic Aggregation at ToR Switch

Require: $timeout \leftarrow timeoutParameter$
Require: $maxlength \leftarrow maxBurstLengthParameter$
{**timeoutParameter and maxBurstLengthParameter parameters are required during ToR switches configuration.** }
Require: $timeoutevent \leftarrow NULL$
Require: $burst\_length \leftarrow 0$
{**Above two lines initialize** $timeoutevent$ **and** $burst\_length$ **parameters.**}
1: **if** $timeoutevent$ OR $burst\_length \geq maxlength$ **then**
{**Above line checks for mixed timer/length based algorithm.**}
2: $control\_packet \leftarrow generateControlPacket()$
{**Above line generates a control packet.**}
3: $control\_packet.setBurstLength(burst\_length)$
4: $control\_packet.setSrcId(getSrcID())$
5: $control\_packet.setDestId(getDestID())$
6: $control\_packet.setSrcAdd(getSrcAdd())$
7: $control\_packet.setDestAdd(getDestAdd())$
{**Above 5 lines set burst length, source and destination IDs, and source and destination IP addresses in the control packet.**}
8: $send(control\_packet, T_{ctrl})$
{**The control packet is sent to the management network after filling required fields.**}
9: **else**
{**Following block is executed when a packet arrives at the VOQ.**}
10: $pk \leftarrow packet\ arrives$
11: **if** $VOQ.empty()$ **then**
12: $firstpk\_time \leftarrow current\_time$
13: $schedule(firstpk\_time + timeout, timeoutevent)$
{**Schedule timeoutevent by adding timeout parameter in first packet's arrival time.** }
14: **end if**
15: $burst\_length += pk.length$
16: $VOQ.insertPacket(pk)$
{**Add packet in virtual output queue.** }
17: **end if**

and is represented by T_{oh}. The T_{oh} includes its propagation delay, O-E-O conversion delay, processing and queuing delay at electrical switch and its transmission delay. The controller processes the control packet and assigns a timeslot on an optical switch path. The time, the controller takes to process the control packet is denoted by T_{proc}. The control packet is sent back to the ToR switch and it again takes T_{oh} to arrive at the ToR switch. The time difference between when it arrives back at the ToR switch and when

Algorithm 6 Burst Transmission at ToR Switch

Require: $timeout \leftarrow timeoutParameter$
{**timeoutParameter is required during ToR switches configuration.**}
1: $cp \leftarrow control\ packet\ arrives$
2: $burstsize \leftarrow cp.getBurstLength()$
3: $portno \leftarrow cp.getPortNo()$
4: $starttime \leftarrow cp.getStartTime()$
{**Above four lines initialize different variables when a control packet arrives at the ToR switch after being processed by the controller.**}
5: $burst \leftarrow generateBurst()$
6: $length \leftarrow 0$
{**Above two lines generate a new burst and initialize its length with zero value.**}
7: **while** $VOQ.hasPackets()$ **do**
8: **if** $length \leq burstsize$ **then**
9: $burst.add(VOQ.getPacket())$
10: $length+ = burst.length$
11: **else**
12: $break$
13: **end if**
14: **end while**
{**Above block inserts packets from VOQ into the generated burst according to its size.**}
15: $sendAt(burst, portno, starttime)$
{**Above line sends the burst on the assigned port at the start time assigned by the controller.**} {**Following steps reinitialize variables after burst transmission.**}
16: **if** $VOQ \rightarrow empty()$ **then**
17: $burst\_length \leftarrow 0$
18: $firstpk\_time \leftarrow 0$
19: **else**
20: $pk \leftarrow VOQ.get(0)$
21: $firstpk\_time \leftarrow pk.arrivaltime$
22: $schedule(firstpk\_time + timeout, timeoutevent)$
23: $burst\_length \leftarrow VOQ.getTotalPacketsLength()$
24: **end if**

it was departed from the ToR switch is called RTT. After processing the control packet, a configuration message is also generated by the controller to configure the optical switch. The configuration message also takes T_{oh} to reach at the optical switch. T_{sw} is the time that a switch takes to configure an optical switch path and is called switch configuration time. In the end, the burst is transmitted at the assigned timeslot on an optical switch path. The time burst takes for the transmission is denoted by T_{tran}. The

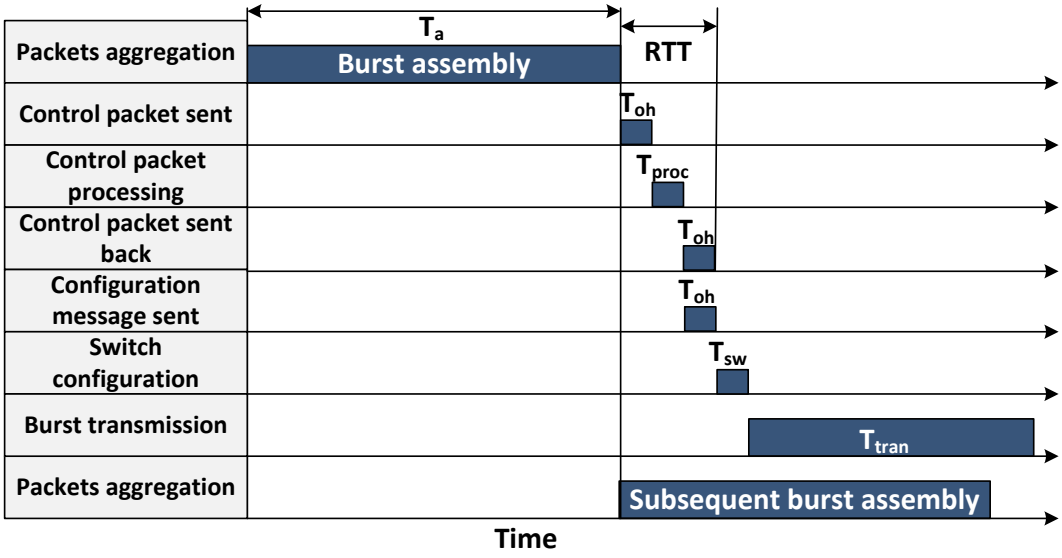


Figure 5.2. Burst Assembly Cycle.

length of the T_{tran} depends upon the size of the burst and data rate of the channel. As the datarate increases, the length of the T_{tran} decreases for the same size of burst. As soon as the control packet is sent by the ToR switch, subsequent burst assembly process is also started and this cycle repeats as long as there is traffic.

5.2.1 Control Plane Processing

The controller keeps a record of the connections of all optical switches. It performs routing, scheduling and switch configuration operations. These operations are depicted in Algorithm 7. There are two data structures which are used to maintain record of horizons of input and output ports (lines 1-2). The controller gets source and destination IDs of ToR switches from the control packet that arrives at the controller (lines 3-5 in Algorithm 7). The controller performs routing operation by finding a minimum value in input and output horizons (lines 6-30 in Algorithm 7). In this method, two values of global horizons one each for input and output ports are initialized to a maximum value (lines 6-7 in Algorithm 7). There are two inner loops that calculate the input and output horizons i.e. lines 11-16 and lines 17-22 in Algorithm 7. This

is a simple operation to get a minimum value. After this, maximum value of these input/output horizons is assigned to both global input/output horizons (lines 23-28 in Algorithm 7). This procedure continues until all optical switches are traversed i.e. outer loop in line 8 in Algorithm 7. The routing operation results in finding optimal input/output ports and their relevant horizons.

Scheduling is the next operation that assigns a timeslot on the selected input/output ports (lines 32-46 in Algorithm 7). The length of the timeslot is calculated from the burst length field in the control packet (lines 35-36 in Algorithm 7). The T_{start} and T_{end} represent the start and end time of the timeslot (lines 37-38 in Algorithm 7) respectively. The T_{sw} is the switching time of the optical switch, T_{proc} is the processing time of the control packet at the controller, T_{oh} is the aggregate time that a control packet spends in the control plane as discussed earlier. A guard time T_{guard} in the timeslot is also considered to avoid synchronization problems. The horizons on the selected input and output ports are updated with a new time (lines 39-40 in Algorithm 7). The controller updates the control packet by assigning a start time and port number on which the burst will be sent (lines 41-42 in Algorithm 7). It then swaps the source and destination IP addresses in the control packet and sends it back to the source ToR switch (lines 43-46 in Algorithm 7).

Switch configuration is the final task of the controller. After processing the control packet, a configuration message is generated (line 47 in Algorithm 7). The controller sets fields such as input port, output port and the time at which a switch will be configured. It also fills source IP address of the controller and destination IP address of the optical switch. In the end, the configuration message is sent to the switch controller for optical switch configuration. The switch controller configures the optical switch according to the instructions in the configuration message.

Algorithm 7 Control Plane Processing for FOSA.

1: $horizoninput[N \times K]$
2: $horizonoutput[N \times K]$
{**Above lines represent data structures of horizons for all inputs and outputs in optical switch paths.**}
3: $controlpacket \leftarrow control\ packet\ arrives\ at\ the\ controller$
4: $srcID \leftarrow controlpacket.getSrcId()$
5: $destID \leftarrow controlpacket.getDestId()$
{**Above lines get source and destination IDs of ToR switches from the control packet arrives at the controller.**}
6: $minInputHorizon \leftarrow maxValue$
7: $minOutputHorizon \leftarrow maxValue$
{**Above two lines represent global input/output horizons which are initialized with maximum value.**}
8: **for** $i = 0$ to $P-1$ **do**
9: $min1 \leftarrow maxValue$
10: $min2 \leftarrow maxValue$
11: **for** $j = i + srcID \times K$ to $(i + srcID \times K + Q - 1)$ **do**
12: **if** $horizoninput[j] < min1$ **then**
13: $min1 \leftarrow horizoninput[j]$
14: $port1 \leftarrow j$
15: **end if**
16: **end for**
17: **for** $k = i + destID \times K$ to $(i + destID \times K + Q - 1)$ **do**
18: **if** $horizonoutput[k] < min2$ **then**
19: $min2 \leftarrow horizonoutput[k]$
20: $port2 \leftarrow k$
21: **end if**
22: **end for**
23: $min3 \leftarrow getMax(min1, min2)$
24: **if** $min3 < minInputHorizon$ **AND** $min3 < minOutputHorizon$ **then**
25: $minInputHorizon \leftarrow min1$
26: $minOutputHorizon \leftarrow min2$
27: $inputport \leftarrow port1$
28: $outputport \leftarrow port2$
29: **end if**
30: **end for**
{**Above blocks of code select optimal input and output ports and their horizon in optical switch path.** }
31: $T_{start} \leftarrow getMax(minInputHorizon, minOutputHorizon)$ {**It gets maximum of two horizons and assigns it to the start time**}
32: **if** $T_{start} < getCurrentTime()$ **then**
33: $T_{start} \leftarrow getCurrentTime()$
34: **end if**
{**Current time is assigned to the start time if horizons are less than current time.**}

35: $burstlength \leftarrow controlpacket.getBurstLength()$
36: $T_{RL} \leftarrow burstlength * 8/datarate$
{**Requested timeslot T_{RL} is calculated from the burst length (BL) in the control packet.**}
37: $T_{start} \leftarrow T_{start} + T_{sw} + T_{proc} + T_{oh}$
38: $T_{end} \leftarrow T_{start} + T_{RL} + T_{guard}$
{**Above lines represent start and end time of a timeslot in an optical switch path.**}
39: $horizoninput[inputport] \leftarrow T_{end}$
40: $horizonoutput[outputport] \leftarrow T_{end}$
{**Horizons are updated with new time.**}
41: $controlpacket.setstarttime(T_{start})$
42: $controlpacket.setport(inputport \ \textbf{mod} \ K)$
{**Control packet is updated with start time and port number of the ToR switch.**}
43: $destadd \leftarrow controlpacket.getsourceadd())$
44: $controlpacket.setdestadd(controlpacket.getsourceadd())$
45: $controlpacket.setsourceadd(destadd)$
46: $sendAt(cp, T_{curr} + T_{proc})$
{**Source and destination addresses in the control packet are swapped and the control packet is sent back to the source ToR. It also completes the scheduling operation. Switch configuration is the next task of the controller.**}
47: $confmsg \leftarrow createConfMsg()$
{**Above line creates a configuration message.**}
48: $confmsg.settime(T_{start} - T_{sw})$
{**Above line sets the time at which switch needs to be reconfigured.**}
49: $confmsg.setinputport((inputport \ \textbf{mod} \ Q) + (srcID \times Q))$
50: $confmsg.setoutputport((outputport \ \textbf{mod} \ Q) + (destID \times Q))$
{**Above two lines set input/output ports on which a connection is configured.**}
51: $confmsg.setdestadd(getOpticalSwitchAdd(\lfloor \frac{inputport \ \textbf{mod} \ K}{Q} \rfloor))$
52: $confmsg.setsourceadd(getControllerAdd())$
{**Above two lines set source and destination address of the configuration message. Lines 47 to 52 describes about the switch configuration operation.**}

5.3 Scalability Analysis of FOSA

Fast optical switch using semiconductor optical amplifiers (SOAs) as a switching fabric with 1024 ports has been proposed in [29] while a fabric with 512 ports using arrayed waveguide grating routers (AWGRs) as a switching fabric is also feasible [60]. Table 5.1 contains a scalability analysis of the proposed topology using both AWGRs and SOAs as the fast optical switches. In Table 5.1, S_{RK} represents the number of servers

Table 5.1. Scalability Analysis for FOSA

Type	*Conf.*	S_{RK}	T_{RK}	*Servers*
SOA	$[1024 \times 1024]$	40	1024	40960
		80	2048	81920
		240	6144	245760
AWGR	$[512 \times 512]$	40	512	20480
		80	1024	40960
		240	3072	122880

per rack while T_{RK} denotes the total number of racks in the DCN. Using SOAs in the switching fabric and ToR switches at its edges, a system size of 40960 servers with 40 servers per rack or 81920 servers with 80 servers per rack can be achieved without requiring a multi-stage topology. If a pod switch is considered instead of the ToR switch that has the capacity to integrate several ToR switches into a single unit and can aggregate a few hundreds to thousand servers [23], leads to the scalability upto 245760 servers by considering 240 servers per pod. A system size of 122880 can also be achieved using AWGRs as a switching fabric by considering 240 servers per pod.

Supporting larger number of servers with current AWGR or SOA technology would require the use of a multi-stage architecture, which would adversely affect latency and blocking probabilities, and therefore limit throughput. However, any demand for DCN capacity in the near to mid term will be for less capacity than these limits. After scalability analysis of the FOSA in this section, the next section describes the performance analysis of the FOSA in terms of latency and throughput. The comparison with traditional methods of OBS and with the baseline electrical network is also provided in the next section.

5.4 Performance Analysis

To assess the performance of OBS for data centre network, simulations models developed for HOSA with TDS in the OMNeT++ were modified to incorporate the functionality of the new technique. In this section, simulated model of the FOSA with

important simulation parameters are presented first, then traffic generation and simulation scenarios are discussed.

5.4.1 Network Topology

The simulation topology consists of 40 ToR switches. Each ToR switch has 40 servers connected to it. The controller and ToR switches are connected to the management network via an electrical switch. The simulation topology uses one fast switch which is interfaced to the management network. The two different cases of network oversubscription ratios i.e. $1:1$ and $2:1$ are considered to investigate the impact of traffic aggregation on the performance of the system. In a fully subscribed network ($1:1$), all servers in a rack send their traffic to servers in other racks, i.e., 100% of the traffic is inter-rack while in a $2:1$ oversubscribed network, 50% of the traffic is inter-rack and 50% of it is intra-rack.

5.4.2 Traffic Generation

In this chapter, instead of using Markov Chain process model for traffic generation, *Weibull* distribution for the inter-arrival rate of the packets is considered to evaluate the performance at 100% offered load. Using Markov Chain, the performance at 100% offered load cannot be measured due to the OFF time. The *Weibull* distribution also represents a good model for the data centre network traffic [121]. Various exponential inter-arrival rates of packets are considered during the ON period to investigate traffic at different loads.

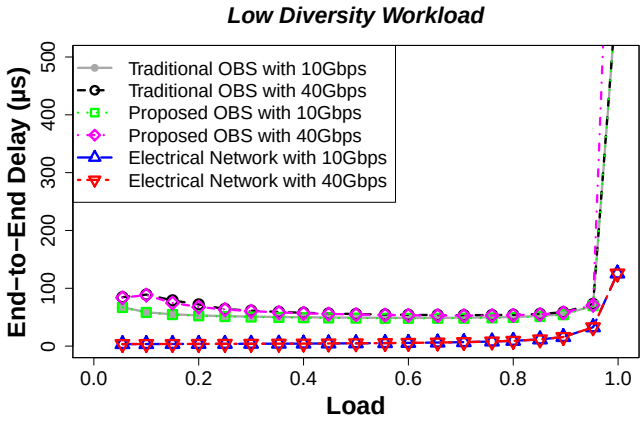
To represent diversity of traffic workload, again the term TDC is used. The TDC is the number of simultaneous destination ToR switches that a given source ToR switch sends traffic to. The different values of TDC parameters are used to evaluate performance at low, medium and high traffic diversity.

Table 5.2. Simulation Parameters for FOSA

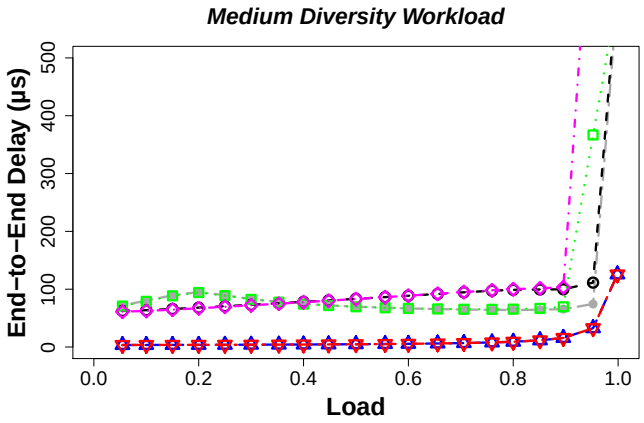
Parameter Name	Symbol	Value
Racks/ToR Switches	T_{RK}	40
Servers per rack	S_{RK}	40
Fast Optical Switch		1
Electrical Switch for control plane		1
Degree of ToR Switches	X	{20,40}
Control packet processing time	T_{proc}	1 μs
Switching Time of Fast Switch	T_{sw}	1 μs
Overhead	T_{oh}	1 μs
Edge to core data rate		{10,40} Gbps
Burst Assembly	T_a	{100 μs,100 KB},{100 μs,400 KB}
Topological Degree of Communication	TDC	$\{1, 10, 20\}$ $Racks$
Data rate from servers to ToR and for control plane		10 $Gbps$
Buffer size per port / VOQ		1000 $packets$

5.4.3 Simulation Parameters

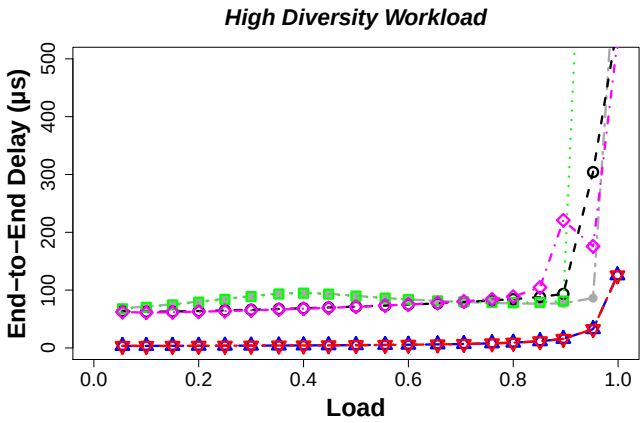
The key simulation parameters are presented in Table 5.2. The simulation models use a value of 1 μs for the switching time of the optical switch. This is a conservative choice, since in some types of fast optical switches, this value can be as low as few nanoseconds [27, 29]. The RTT of the control packet includes its processing time at the controller (T_{proc}) and twice of the overhead time (T_{oh}). The aggregate value of T_{oh} is conservatively set to 1 μs although all these delays are negligible (at most a few nanoseconds [54]). A value of 1 μs for T_{proc} is used. It is compatible with its actual value that is measured in the next section. In our recent work [132], we investigate the performance of OBS for data centres using various burst assembly parameters. Herein, optimum values are used i.e. {100 μs,100 KB} for 10 $Gbps$ and {100 μs,400 KB} for 40 $Gbps$ data rates. Three values of the $TDC = \{1, 10, 20\}$ are used to investigate the impact of traffic diversity on the performance of the system. The simulation models consider a buffer size of 1000 packets (i.e. 1.5 MB) per port / VOQ while state of the art ToR switches can support a higher buffer size [103, 104]. The minimum value of buffer size should be greater than the maximum burst size (i.e. 100KB at a 10 Gbps data rate and 400KB at a 40 Gbps data rate).



(a)

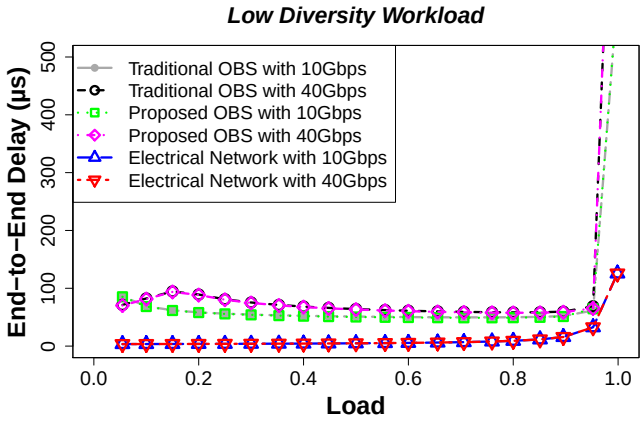


(b)

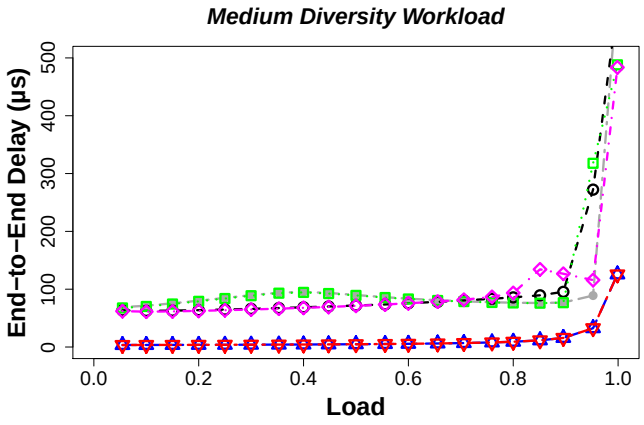


(c)

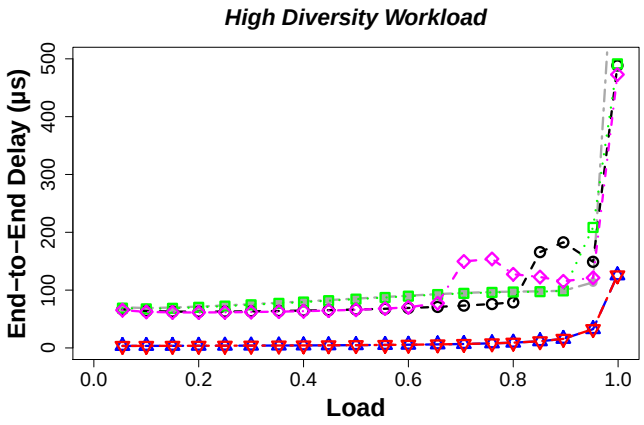
Figure 5.3. Load Vs End-to-End Delay measured in the fully subscribed network for:(a) TDC = 1, (b) TDC = 10 and (c) TDC = 20.



(a)



(b)



(c)

Figure 5.4. Load Vs End-to-End Delay measured with 2:1 oversubscribed network for: (a) TDC = 1, (b) TDC = 10 and (c) TDC = 20.

5.5 Results and Discussion

The performance of OBS is examined by measuring latency and throughput. The comparative analysis of the proposed design using *OBS with two-way reservation* with

OBS using traditional methods of one-way reservation is provided. The comparison is also done with a Traditional Electrical (TE) packet switching network that features a two layer leaf-spine topology [131] as shown in Figure 4.6 from Chapter 4. Its latency and throughput performance provides a baseline against which the performance of the new networks can be benchmarked. The TE network acts as an ideal electrical packet switching network that has low latency and high throughput. The simulation results obtained are shown in Figures 5.3, 5.4, 5.5 and 5.6. This chapter also discusses the performance of the algorithms in the control plane later in this section.

5.5.1 Latency

The term end-to-end delay is used to measure the latency. The latency of only inter-rack traffic is investigated so that the performance of optical interconnect could be evaluated. The latency of intra-rack traffic is negligible due to nanoseconds switching time of electrical switches.

The simulation results obtained for latency are shown in Figures 5.3 and 5.4. Figure 5.3 deals with the delay performance at different values of offered load by considering three values for TDC in the fully subscribed network while Figure 5.4 describes the delay performance at different values of offered load by considering three values for TDC in $2:1$ oversubscribed network. First two curves at each plot in Figures 5.3 and 5.4 represent end-to-end delay versus offered load using OBS with traditional methods of one way reservation scheme using 10 and 40 $Gbps$ data rates respectively. Third and fourth curves represent performance of the proposed methods of OBS using two-way reservation scheme while the last two curves show the corresponding performance of the baseline electrical network using 10 and 40 $Gbps$ data rates. It can be seen that delay performance in the proposed scheme of OBS is compatible with the delay performance of traditional methods of OBS i.e. effect of additional delay caused by two-way reservation in the proposed scheme is negligible across all cases of different workloads. However, the delay is slightly higher in the traditional and the proposed OBS as compared to delay in the baseline electrical network. This is due to effect of burst assembly delay at the ToR switches. This delay results from the

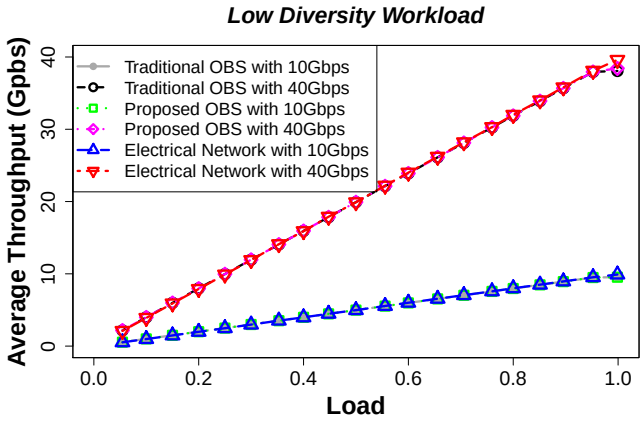
limitation of OBS but is still comparable with the baseline electrical network.

In the baseline electrical network, the delay is around 10 μs until 90% load is reached and it increases thereafter. In a traditional OBS, the delay is around 50-80 μs up to a high load and increases thereafter. The delay performance of the proposed OBS is similar. The additional delay in OBS is due to the effect of burst assembly delay at the ToR switches, which is an inherent limitation of OBS. Nonetheless, such a delay is acceptable for most HPC applications [133]. HPC applications can be categorized into one of three categories: (1) Tightly coupled applications, (2) Loosely coupled applications and (3) Parametric execution applications [134]. These applications are characterized by their significant interprocessor communication (IPC) message exchanges among the computing nodes. The tightly coupled applications are very latency sensitive and require a latency of at most tens of microseconds. In loosely coupled applications, the applications in this category involve little or no IPC traffic among the computing nodes. So low latency is not a requirement. Similarly, the parametric execution applications are also latency insensitive due to there being no IPC traffic.

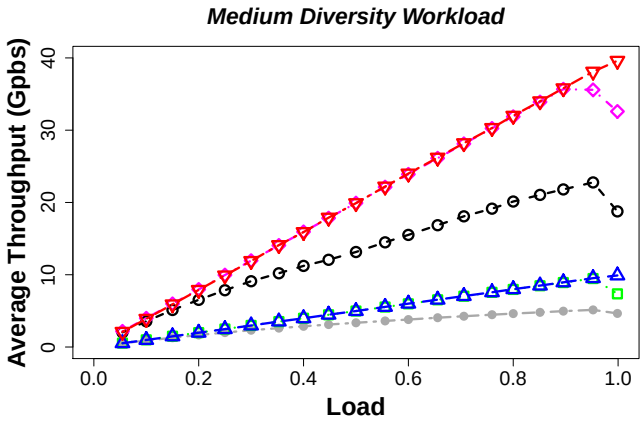
5.5.2 Throughput

The simulation results obtained for throughput are shown in Figures 5.5 and 5.6. Figure 5.5 presents the throughput performance at different values of offered load by considering three values for TDC in fully subscribed network while Figure 5.6 exhibits the throughput performance at different values of offered load by considering three values for TDC in 2 : 1 oversubscribed network. First two curves at each plot in Figure 5.5 and 5.6 represent average throughput per link versus offered load using OBS with traditional methods of one way reservation scheme using 10 and 40 $Gbps$ data rates respectively. Third and fourth curves represent performance of the proposed methods of OBS using two-way reservation scheme while the last two curves show the corresponding performance of the baseline electrical network using 10 and 40 $Gbps$ data rates.

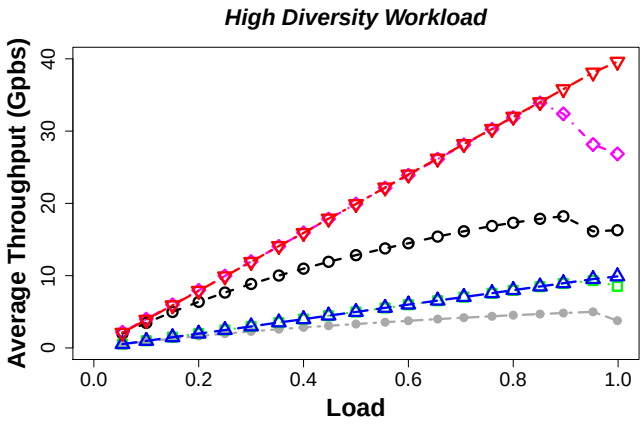
It can be noticed in Figures 5.5(a) and 5.6(a) that the average throughput is identical across all the networks with low diversity traffic workload. This is because the



(a)



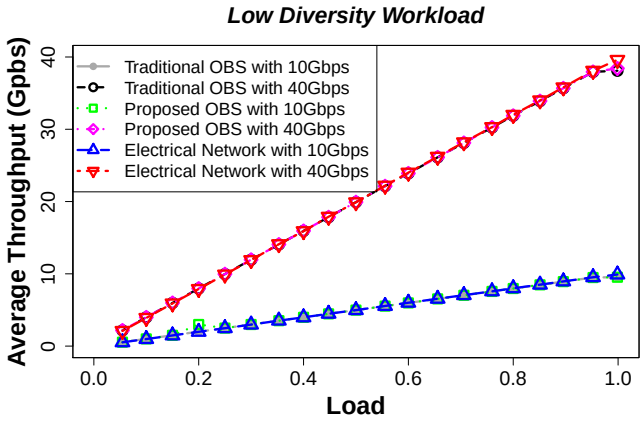
(b)



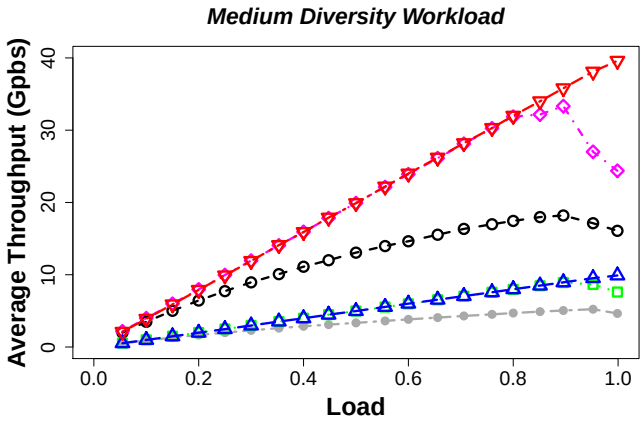
(c)

Figure 5.5. Load Vs Average Throughput measured in the fully subscribed network for: (a) TDC = 1, (b) TDC = 10 and (c) TDC = 20.

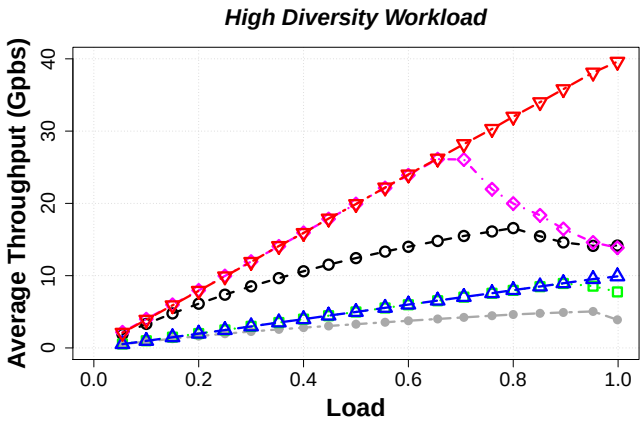
burst loss in OBS with traditional methods is also zero in this case. The bursts in each ToR switch are generated in an order and all these bursts are destined to only one ToR



(a)



(b)



(c)

Figure 5.6. Load Vs Average Throughput measured with 2:1 oversubscribed network for: (a) TDC = 1, (b) TDC = 10 and (c) TDC = 20.

switch. So there is no overlap/contention which results in zero burst loss. However, packets are lost at a very high load i.e., at 95% load due to buffer overflow at ToR

switches. Packet losses also occur in the proposed scheme as well as in the baseline electrical network due to buffer overflow at a very high load. Because of this, the average throughput decreases slightly across all networks at high load.

In medium diversity workload as shown in Figures 5.5(b) and 5.6(b), the average throughput in traditional methods of OBS is considerably low as compared to the average throughput achieved in the proposed scheme. This is because, bursts are lost in the traditional methods of OBS due to contention while in the proposed scheme burst loss is zero due to the usage of two-way reservation protocol. The proposed scheme also demonstrates performance comparable to that of the baseline electrical network till a very high load. Another important point is that the average throughput decreases with the increase of data rate at a very high load. This is because bandwidth is wasted during assignment of the timeslot in a link. This wasted bandwidth is four times high using 40 $Gbps$ as compared to 10 $Gbps$ data rates. A similar trend of decrease in average throughput is also observed with high diversity workloads as shown in Figures 5.5(c) and 5.6(c).

It is worth noticing the impact of network oversubscription on the average throughput in Figures 5.5(b) and 5.6(b). The drop in average throughput in the proposed scheme with 40 $Gbps$ in the fully subscribed network is less as compared to $2:1$ oversubscribed network. A similar trend is observed in Figures 5.5(c) and 5.6(c). This is because, more links are available in the fully subscribed network as compared to a $2:1$ oversubscribed network. So the chances of getting a timeslot are high in the fully subscribed network as compared to $2:1$ oversubscribed network. This ultimately results in getting a high average throughput in the fully subscribed network.

5.5.3 Packet Loss Ratio

The simulation results obtained for the packet loss ratio are shown in Figure 5.7 for different values of offered load. Two values are considered for TDC in the fully subscribed network. The first two curves in each plot in Figure 5.7(a) and 5.7(b) show packet loss ratio as a function of offered load for OBS using traditional methods of one way reservation scheme at data rates of 10 and 40 Gbps respectively. The third

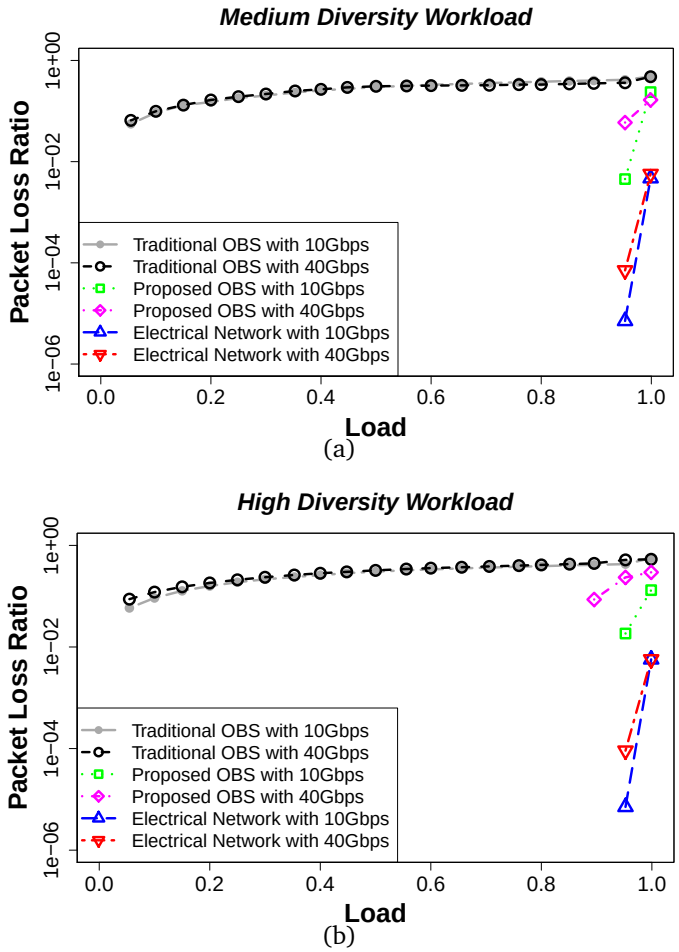


Figure 5.7. Load Vs Packet Loss Ratio measured in the fully subscribed network for: (a) TDC = 10 and (b) TDC = 20.

and fourth curves show the equivalent performance of the proposed methods of OBS using a two-way reservation scheme, while the last two curves show the corresponding performance of the baseline electrical network. Packet losses are observed in the proposed and the baseline network only at a very high load while the packet losses in traditional OBS occur even at very low load. This is due to burst losses caused by contention in a traditional OBS. Similar results may be observed in a 2:1 oversubscribed network, although these are omitted from Figure 5.7 for clarity.

5.5.4 Performance of the Control Plane

In order to assess the performance of the control plane, the algorithm was run on an Intel host with a Core i7, 2.17 GHz processor and 16 GB RAM. The results were

Table 5.3. Performance of the Control Plane in FOSA

Algorithm	*Over-subscription*	*Optical Switch (P)*	*Degree of ToR(X)*	*Exec.T*
Routing and scheduling	4 : 1	$\forall P$	10	$< 0.15\ \mu s$
	2 : 1		20	$< 0.3\ \mu s$
	1 : 1		40	$< 1\ \mu s$
Switch Configuration	4 : 1	10	10	$\approx 0.029\ \mu s$
	2 : 1	20	20	$\approx 0.031\ \mu s$
	1 : 1	40	40	$\approx 0.033\ \mu s$

obtained for several combinations of parameters. To ensure statistical significance, the results of 1000000 runs were averaged. The results are shown in Table 5.3.

When a control packet arrives at the controller, the controller performs the routing, scheduling and switch configuration operations described in Algorithm 7. The routing and scheduling operations are described from line 1 to 46 and switch configurations operations are described from lines 47 to 53 in Algorithm 7. The complexity of the routing and scheduling algorithm is $O(2X + \mu)$, where X is the degree of ToR switches and μ represents the sum of processing time of all other instructions. The complexity of the switch configuration operations is $O(P + \mu)$ where P is the total number of optical switches. The μ is assumed to be a constant of negligibly low value and its value is in the range of a few nanoseconds. The execution time is measured in the fully subscribed, 2 : 1 oversubscribed and 4 : 1 oversubscribed networks as shown in Table 5.3. 40 servers per rack are considered. It can be noticed in Table 5.3 that the execution time of routing and scheduling operations is in nanoseconds scale for all types of networks. It is minimum in 4:1 oversubscribed network but it increases slightly as we decrease network oversubsription. Similarly, the execution time of the switch configuration operations is minimum when P is minimum and it increases slightly with the increase of the number of optical switches. The overall execution time of switch configuration operations is negligible (at most a few nanoseconds). We get a total execution time of the control plane processing by adding up the execution times of routing/scheduling and switch configuration operations which is in nanoseconds range. So the proposed algorithms in the control plane demonstrate efficient performance across all types of network oversubscriptions.

5.6 Conclusion

In this chapter, a novel optical interconnect based on fast optical switches called FOSA was studied. The previous hybrid design HOSA with TDS has the limitation of the control plane. The control plane can only support applications that have high traffic stability, i.e. workloads that last several seconds. So for dynamically changing traffic patterns, the new architecture FOSA features fast optical switches in a single hop topology with a centralized, optical control plane. Similar to the HOSA, the single stage core topology can be easily scaled up and scaled out.

The OBS with two-way reservation is considered to get zero burst loss. The two-way reservation is not appropriate for conventional backbone optical networks due to the high RTT of the control packet but in a DCN, this RTT is not high. The network-level simulation is used to model different workloads with various data rates by considering different edge to core network over-subscription and investigate the performance of such designs across various usage patterns. The results reveal that the proposed technique shows considerable improvement in terms of throughput and packet loss ratio as compared to the conventional methods of OBS while comparable performance in terms of delay with the conventional methods of OBS is also achieved. The proposed technique also demonstrates delay and throughput performance comparable to that of electrical data centre networks.

Apart from the better performance of the FOSA than the OBS using traditional methods and comparative performance with the baseline electrical network, it also has better performance than the HOSA and HOSA with TDS. Unlike HOSA with TDS, the FOSA has efficient control plane performance as well. This is because no additional overhead for maintaining statistics of traffic demand is required in the FOSA. In the next chapter, the performance of TCP in the FOSA is evaluated and its comparison with the traditional OBS and the baseline electrical network is provided.

CHAPTER 6

PERFORMANCE EVALUATION OF TCP OVER FAST OPTICAL SWITCH ARCHITECTURE FOR DCN

6.1 Introduction

In this chapter, the performance of TCP over FOSA is evaluated for use in a DCN using network-level simulation. The performance of TCP over OBS networks is degraded because of the wrong interpretation of congestion in the network by its flow control algorithm. The contention induced losses are responded to as if they were congestion induced losses. The former are burst losses occurring due to unavailability of a wavelength even at the low network load.

To evaluate the performance of TCP over FOSA, various workloads with different burst assembly parameters are used to compare TCP performance with two-way reservation to its performance with conventional methods of one-way reservation and to its performance in a conventional electronic packet switching DCN.

6.2 TCP over OBS

The basic TCP protocol is presented in RFC793 [135], but there are many RFCs [136, 137] which have introduced extensions to the TCP to improve its performance. The TCP sender sends data by breaking it into small chunks which are called segments. The receiver receives the segment and sends back an acknowledgement (ACK) to the sender. The time between sending of the segment and the receipt of its acknowledgement is called the RTT. This RTT is different from the RTT of the control packet in OBS. Flow control and congestion control are the two main features of TCP. The flow control is imposed by the receiver using a receiver window, that indicates the amount of data it can receive due to memory limitation and the congestion control is implemented in the sender using a congestion window, that is a limit on the amount of data the sender can transmit into the network before receiving an ACK in order not to overload the network. The sender can send at most the maximum of the receiver window or the congestion window bytes to the network. Another important feature of TCP is reliability. The TCP sender detects the loss of a segment by means of the reception of three duplicate ACKs, or by the triggering of a Retransmission Timeout (RTO).

TCP uses a Slow Start and Congestion Avoidance mechanism to update the congestion window. In Slow Start, the TCP congestion window size is equal to one maximum segment size (MSS) and its size is doubled after receipt of the ACK until a threshold is reached. When a packet loss is detected, the congestion window is reset to 1 MSS and the Slow Start is started again until the congestion window reaches a certain limit (Slow Start threshold), at which time the Congestion Avoidance mechanism is applied. In this mechanism, the window is incremented by 1 MSS per RTT.

Burst loss and delay caused by the burst assembly and FDLs are the important features of OBS that have the most impact on TCP performance. The burst loss can be misinterpreted as congestion in the network instead of contention. The timeout triggered by the contention is termed False Time Out (FTO). After FTO, TCP sender starts with a Slow Start mechanism which ultimately decreases network throughput. In most cases, several packets from different TCP sessions are included in a burst and the burst drop could result in a loss of many packets per session, resulting in a network

wide drop in throughput [138]. Another factor affecting the performance is the delay that a packet experiences during the burst assembly process before its associated burst is transmitted and also in the FDLs ring if the burst is routed through in it during contention. If this accumulative delay is higher than RTO, then TCP senders start again with Slow Start resulting in a decrease of throughput.

TCP over OBS networks also suffers from a problem known as the high Bandwidth Delay Product (BDP). The BDP determines the amount of data that can be in transit in the network i.e. the amount of data that can be sent over the network without being acknowledged. TCP throughput is bounded by the BDP. If the TCP sender window is smaller than the BDP, there is a waste of link capacity, and the TCP sender will be idle most of the time. The TCP window is limited to 64 KB, which is much lower than the BDP in most optical networks.

In order to overcome the limitations of TCP over OBS, many techniques have been presented in the literature [139–148]. The authors in [139] evaluated the impact of burst assembly algorithms on different TCP implementations such as TCP Reno, New-Reno and SACK in OBS network and in other work [140], they proposed a TCP implementation for OBS network called Burst TCP which tries to detect false time out and reacts appropriately. In [141], the authors introduced a burst retransmission scheme in which the bursts lost due to contention in the OBS network are retransmitted at the edge node. High Speed-TCP (HS-TCP) is a modification to TCP's window increase and decrease algorithm that allows it to run efficiently on networks with large BDP [149]. The authors in [142] evaluated the behaviour of High-speed TCP in OBS networks. Another technique [148] proposed modification in the burst assembly period at the edge node that aggregates packets from different TCP sessions into different bursts. In [143], the authors presented a TCP Vegas implementation using a threshold-based mechanism to identify network congestion under burst retransmission scheme. A source-ordering technique over a load-balanced OBS network is introduced by the researchers in [144] to avoid false time out. In [147], the authors presented predictive techniques in OBS that try to improve TCP performance. TCP is very sensitive to the packet/burst losses and this has been shown in an empirical study of different TCP implementations in OBS [146]. The authors in [145] presented a protocol level tech-

nique for estimating assembly time at the TCP end points and use this information to differentiate congestion induced loss from contention induced loss. The authors in [150] introduce a new layer between the TCP and OBS layers for burst retransmission to mitigate the effect of burst loss due to contention on TCP performance. The new layer can handle the buffering, sequencing and retransmission of bursts. It requires modification of ingress nodes to adapt the functionality of a new layer.

We observe through the literature survey that burst loss in TCP over OBS is still a major issue because the techniques proposed so far either require modifications at the protocol level or they are designed to be implemented in ingress nodes which is a difficult task. These techniques can help to improve TCP performance in the OBS network but none of them can entirely avoid zero burst loss due to contention. Due to the sensitivity of TCP to the packet/burst losses, burst losses in the OBS network result in a much lower network throughput than in the equivalent electrical network. The proposed technique uses the two-way reservation protocol to ensure zero burst loss and hence significantly improves the TCP performance. The proposed technique also uses a small burst assembly time (a few microseconds) that is much smaller than the default retransmission timeout, so there is no effect of burst assembly delay on retransmission in the proposed solution.

6.3 Performance Analysis

To assess the performance of TCP over OBS for data centre network, in addition to the simulation models presented in previous chapters, the INET models of OMNeT++ are used here to simulate the behaviour of TCP. Important simulation parameters are presented in Table 6.1. The simulation model consists of 24 ToR switches. Each ToR switch has 40 servers connected to it. The controller and ToR switches are connected to the management network via an electrical switch. One fast optical switch is used which is interfaced to the management network and ToR switches.

A bijective traffic model is considered in which the number of TCP flows a server generates is equal to the number of TCP flows the server receives. 40 TCP flows per

Table 6.1. Simulation Parameters for TCP over FOSA

Parameter Name	**Symbol**	**Value**
Racks/ToR Switches	N	24
Servers per rack	S_{RK}	40
Degree of ToR Switches	X	40
TCP flows per server		40
Data in each TCP flow		25 MB
Topological Degree of Communication	TDC	{1,4,8} Racks
Data rate	R	10 Gbps
TCP Window Size		64 KB
Control packet processing time	T_{proc}	1 μs
Switching Time of Fast Switch	T_{sw}	1 μs
Overhead	T_{oh}	1 μs
Burst Aggregation	T_a	{{50 μs, 50 KB}, {50 μs, 100 KB}, {100 μs, 50 KB}, {100 μs, 100 KB}}
Buffer Size per electronic port / VOQ		1000 packets

server (matching the number of servers in a rack) are used. Each TCP flow in a server sends 25 MB data to another server, so each server sends in total $25 \times 40 = 1GB$ data to other servers. Three cases of topological degree of communication TDC are considered to investigate the traffic diversity workload. The TDC represents rack level flows i.e. TDC=1 means that servers in a rack send data traffic to servers in only 1 destination rack while TDC=4 reveals that each server in a rack sends data traffic to 10 servers each in four racks (total 40 servers in 4 racks). Similar method is used for TDC=8 (i.e. each server in a rack sends 5 TCP flows to 5 servers each in eight racks). Asymmetric traffic is considered, which means that the servers in rack A send data to the servers in rack B, the servers in rack B send data to the servers in rack C while the servers in rack B and C only send ACKs to the servers in rack A and B respectively. TCPReno models are used for TCP implementation which are available in OMNeT++ INET models [117].

The simulations models use a value of $1\mu s$ for the switching time of the optical switches. The aggregate value of T_{oh} is conservatively set to $1\mu s$ although all these delays are negligible (at most a few nanoseconds [54]). A value of $1\mu s$ for T_{proc} is considered. Four cases for traffic aggregation are considered by using values of aggregation drawn from the set {{50 μs,50KB}, {50μs,100KB}, {100μs,50KB}, {100μs,100KB}}.

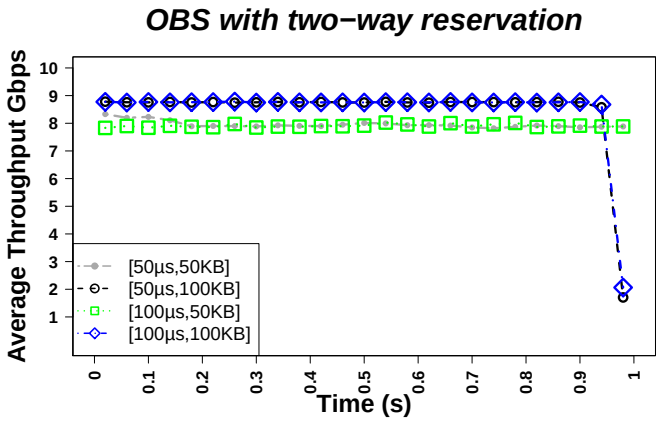
A buffer size of 1000 packets is conservatively selected (i.e. 1.5MB per electronic port / VOQ in ToR switches and switches in electronic DCN), while state of the art switches can support a much higher buffer size [103, 104]. The minimum value of buffer size should be greater than the maximum burst size (i.e. 100KB at a 10Gbps data rate).

6.4 Results and Discussion

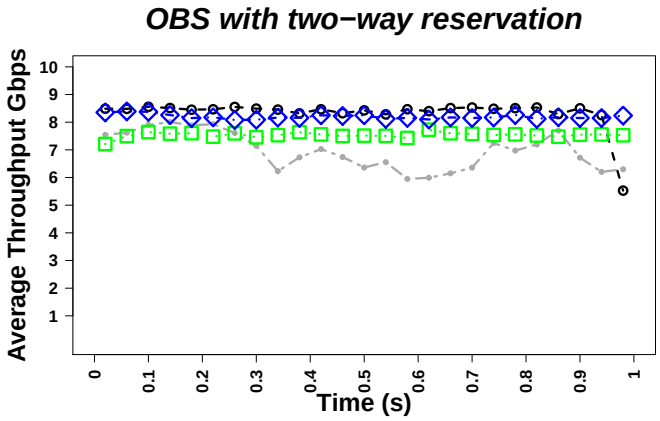
The performance of TCP is examined by measuring throughput, completion time, packets loss and round trip time of TCP segments. The TCP performance of the proposed technique, using OBS with two-way reservation, is compared with OBS using traditional methods of one-way reservation. The TCP performance of a conventional DCN based on electronic packet switching using simulation is also evaluated as a baseline. A two layer leaf spine topology for the electronic packet switching DCN is used. The detailed description of this topology is given in Chapter 5. The simulation results obtained are shown in Figures 6.1 ,6.2, 6.3, 6.4, 6.5 and 6.6.

6.4.1 Throughput

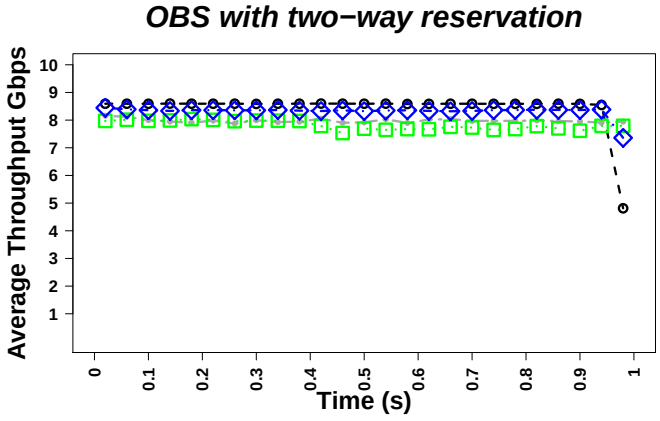
Figure 6.1 shows the throughput performance achieved for three values of TDC across a range of burst assembly parameters in the proposed design while Figure 6.2 shows the throughput performance using one-way reservation. Four curves in each plot of Figures 6.1 and 6.2 represent the average throughput for various values of the burst aggregations. The burst loss of OBS with two-way reservation is zero across all values of TDC, leading to high throughput as shown in all plots of Figure 6.1. In OBS with traditional one-way reservation, the performance is good only when TDC $= 1$. The performance is degraded with the increase of traffic diversity e.g. with TDC $= 4$ and TDC $= 8$ as shown in Figures 6.2(b) and 6.2(c). This is because burst loss in OBS with one-way reservation is negligible when TDC $= 1$ as all bursts are going to only one rack and the requests for burst reservations arrive in an order. But in case of higher TDC values, burst loss increases with increasing TDC because of a large number of overlapping control packet requests. The network throughput is fair initially as shown



(a)



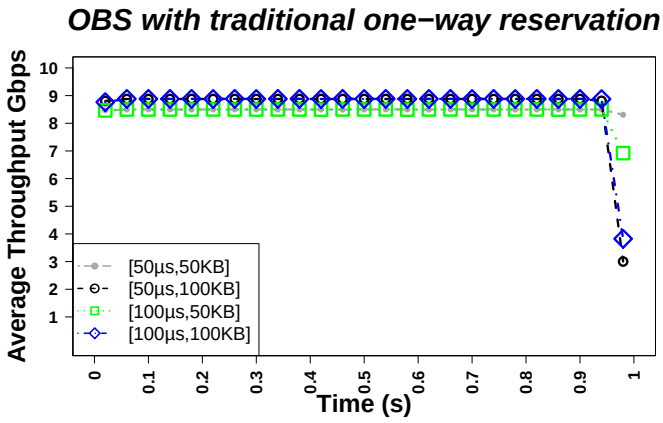
(b)



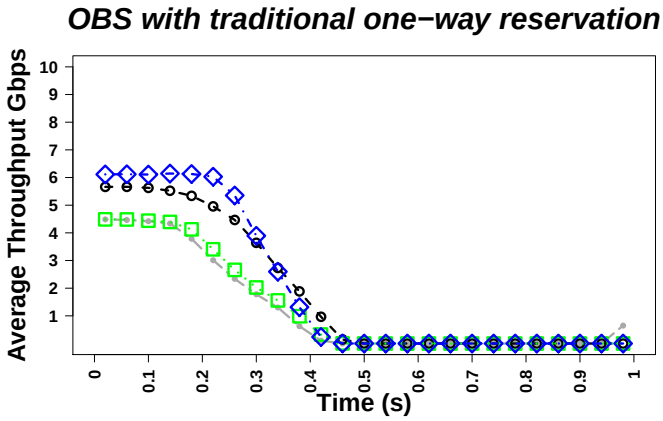
(c)

Figure 6.1. Average throughput of the proposed design using OBS with the two-way reservation by considering different burst aggregation parameters and with respect to different TDC values, (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

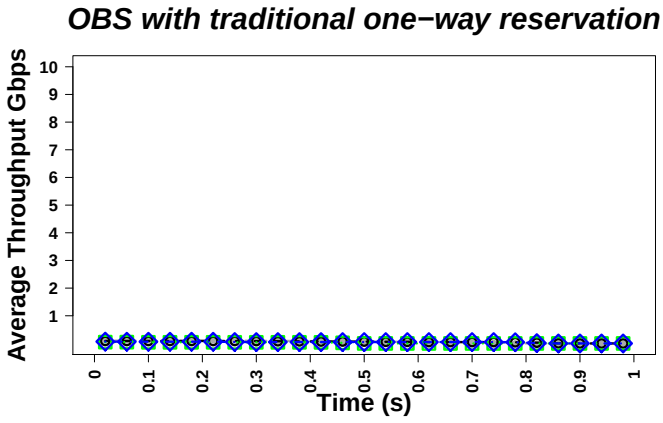
in Figure 6.2(b) but after half of a second, throughput drops and servers go into Slow Start. In the case of TDC = 8 as shown in Figure 6.2(c), the throughput is poor



(a)



(b)



(c)

Figure 6.2. Average throughput of OBS with traditional methods of one-way reservation by considering various burst aggregation parameters and for various TDC values, (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

due to the large number of burst losses. Similar trend of decrease in the throughput with the increase of TDC values is observed in the electronic packet switching DCN as

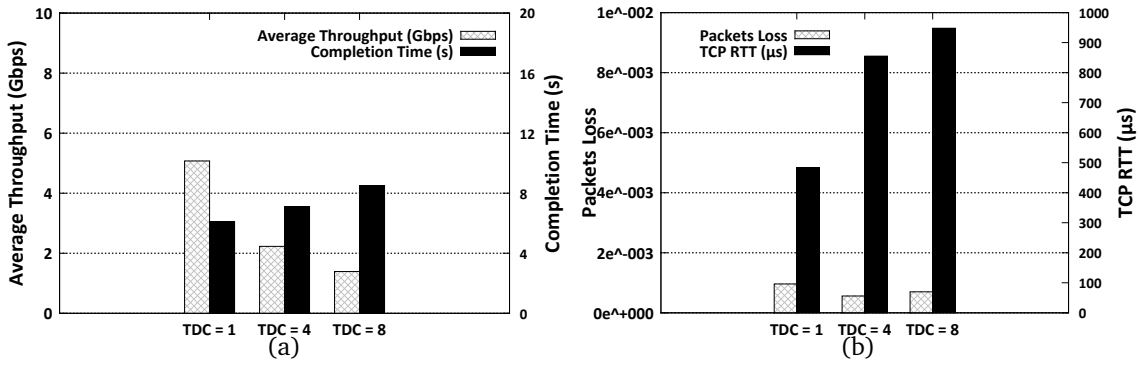


Figure 6.3. Performance analysis of TCP over conventional electronic packet switching DCN for various values of TDC: (a) Average throughput achieved during first second of simulation time and completion time to transfer 1GB data from each server, (b) Packets loss and average round trip time of TCP segments.

shown in Figure 6.3(a). The packets are lost in the electronic network due to buffer overflow both at edge and core nodes which results in a decrease of throughput. The throughput achieved in the proposed scheme is better than traditional methods of OBS and electronic DCN.

6.4.2 Completion Time

Figure 6.4 shows the time taken by all servers to send 1 *GB* data traffic to other servers for three values of TDC across a range of burst assembly parameters in both designs. It can be observed that servers only take a couple of seconds to complete a data transfer in the proposed scheme but in the traditional methods, they take tens of seconds (greater than 20 s) to complete a data transfer. This is because when packets are lost, servers go into Slow Start after Retransmission Timeout. This process continues as long as there are packet losses. Due to repeatedly entering the Slow Start phase, the servers are not able to utilize their full capacity. In the electronic network, the servers take 5-8 seconds to complete the data transfer as shown in Figure 6.3(a) which is better than using traditional methods of OBS but almost three times as compared to the proposed scheme.

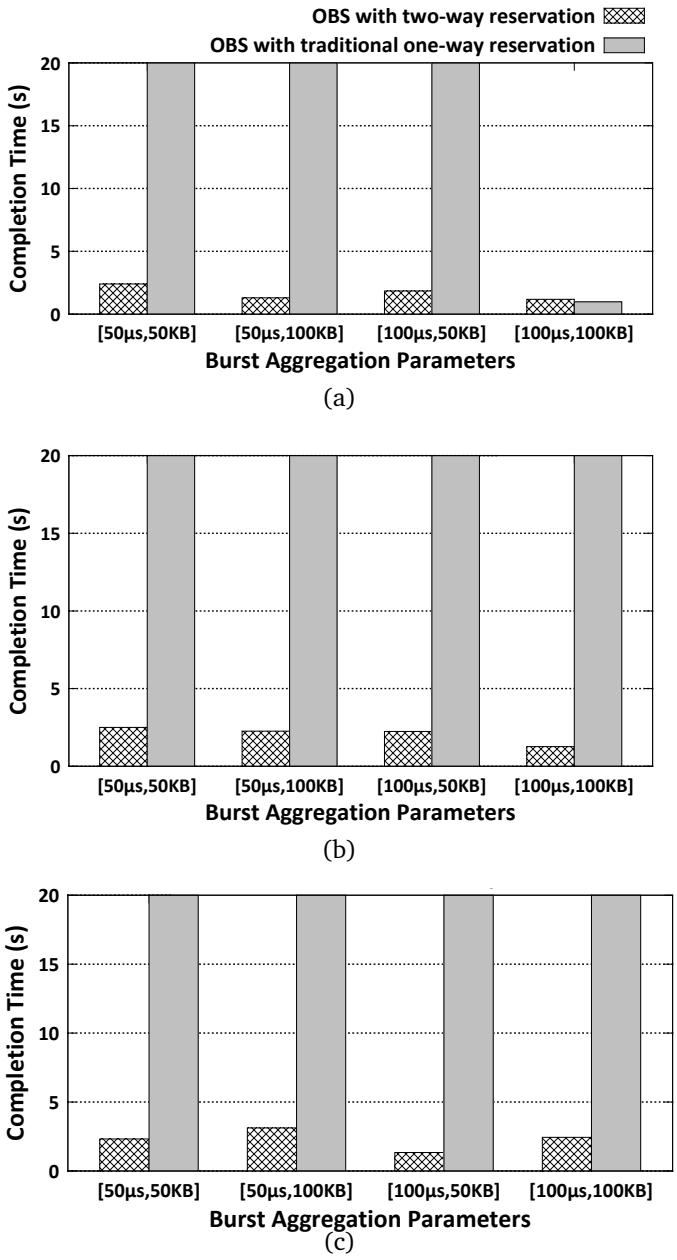


Figure 6.4. Completion time to transfer 1GB data from each server for various burst aggregation parameters and for various TDC values, (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

6.4.3 Packet Loss

Figure 6.5 shows the packet losses for three values of TDC across a range of burst assembly parameters in both designs. Packets can be lost in the proposed design due to buffer overflow in the network interface card (NIC). An NIC buffer capacity equal to 500 packets is considered. For all three values of TDC, the packet loss is negligible in

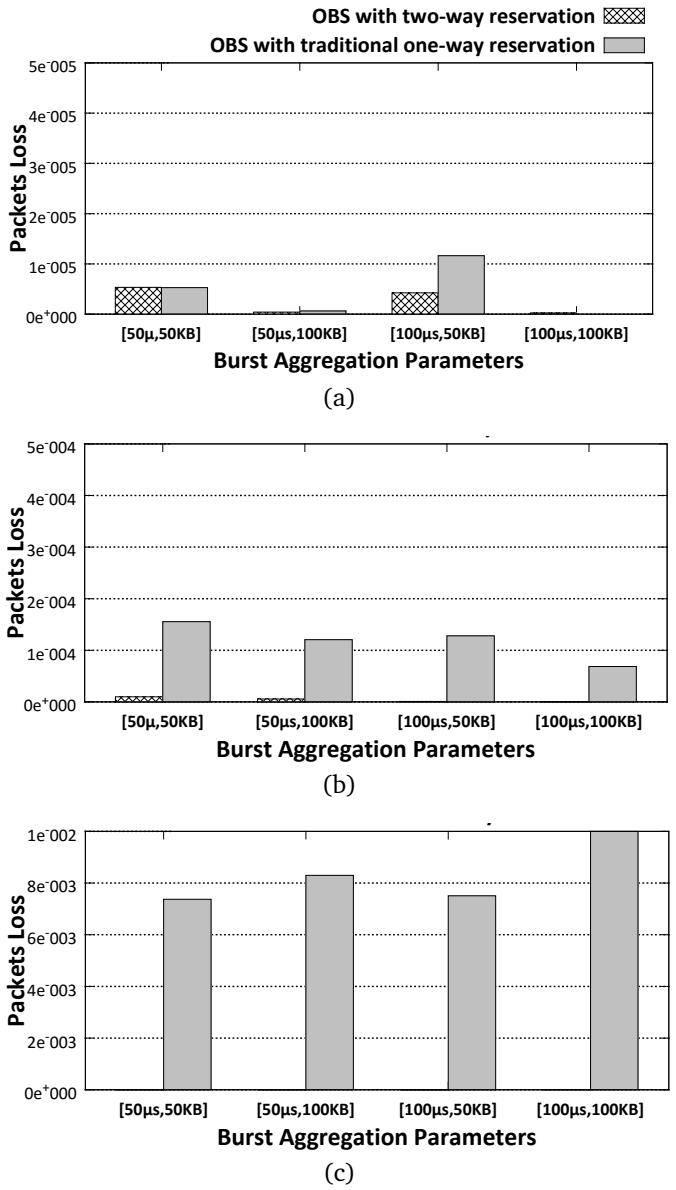
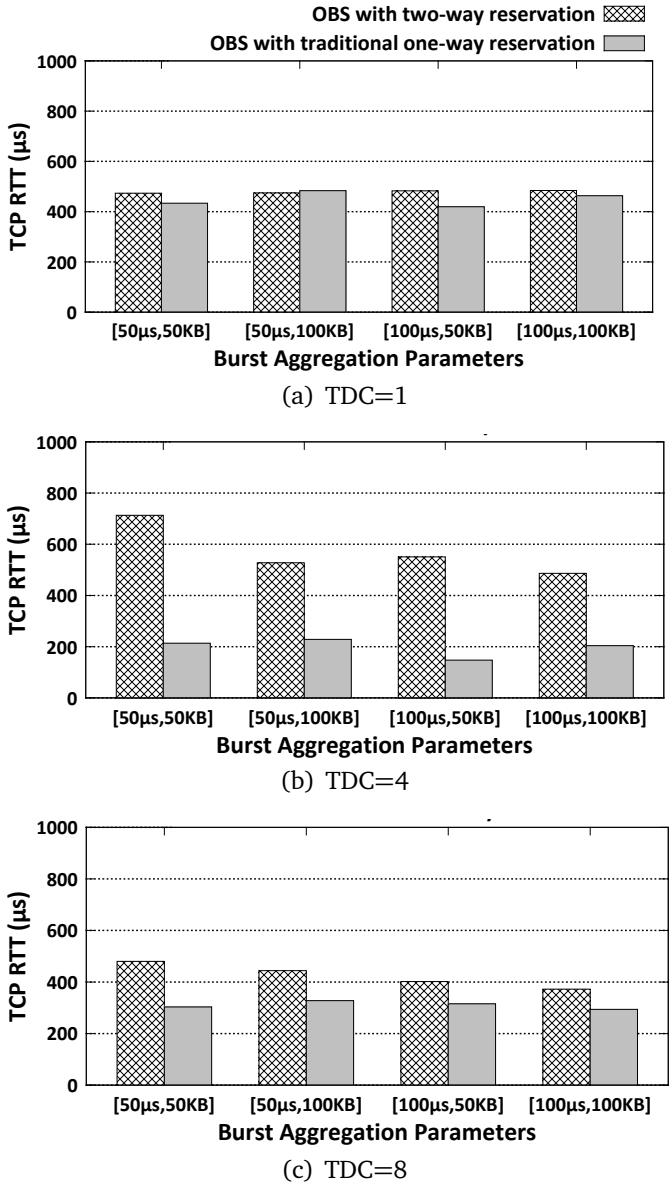


Figure 6.5. Packets loss for various burst aggregation parameters and for various TDC values, (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

the proposed scheme, while in the traditional methods, the packet loss increases with the increase of TDC as shown in Figures 6.5(b) and 6.5(c). This is due to the higher number of burst losses with the high diversity traffic workload. In the electronic DCN, the packet losses are lower than for traditional methods of OBS but are slightly higher than the proposed scheme for all types of workload as shown in Figure 6.3(b).



(a) TDC=1

(b) TDC=4

(c) TDC=8

Figure 6.6. Average round trip time of TCP segments for various burst aggregation parameters and for various TDC values, (a) TDC = 1, (b) TDC = 4, (c) TDC = 8.

6.4.4 Round Trip Time

Figure 6.6 shows the average round trip time (RTT) of TCP segments for three values of TDC across a range of burst assembly parameters in both designs. The round trip time is measured from when a TCP segment is sent by the TCP application in a source server to when its acknowledgement is received at the source server. The RTT in all three cases of TDC across different burst aggregation schemes in the proposed

scheme is around 500 μs. It can be noticed that the aggregation time that the proposed technique considers is only 50 μs and 100 μs while the RTT is around 500 μs. This is because TCP segments also spend some time in the queue of NIC in servers and ToR switches. When there is a high traffic load, these segments have to wait in the queue to get transmitted. This is not the case in the traditional methods of OBS using one-way reservation. For example, with TDC = 4 and TDC = 8 as shown in Figures 6.6(b) and 6.6(c), the RTT in most of the cases is around 200μs because there is not much traffic to send due to a lower throughput observed in the traditional methods of OBS. So, TCP segments find it easy to go through the queue of NIC in low traffic that results in lower RTT. The RTT in the electronic network is higher than in both OBS schemes as shown in Figure 6.3(b). This is because of the additional delay which is incurred at the core node while in traditional and proposed methods of OBS delay is incurred only at the edge node.

The bandwidth delay product in a data centre network is low compared to that of a backbone long haul optical network due to its low RTT. The effect of BDP can be eliminated if we increase the number of TCP flows per server or the window size. By considering a default window size of 64KB and 40 TCP flows per server, good TCP performance comparable to that of the electrical data centre network has been achieved.

6.5 Conclusion

In this chapter, the performance of TCP over an optical burst-switched FOSA for data centre network is investigated by using network-level simulation. Burst loss is the major limitation of traditional OBS that degrades the performance of TCP. Due to the zero burst loss in OBS with two-way reservation, efficient TCP performance is observed.

Various burst assembly parameters were examined with various traffic workloads to evaluate the performance of TCP. The results show a significant improvement in the throughput, completion time and packet loss as compared to the traditional methods of OBS across all types of workloads. The proposed scheme also demonstrates better

TCP performance than the conventional electronic packet switching network for all types of workloads.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

This thesis presents three novel optical interconnection schemes for data centre networks which are based on OBS. OBS was initially proposed for long-haul backbone optical core networks but it has not replaced OCS due to its limitation of high burst loss in this application. The proposed schemes consider OBS with a two-way reservation protocol that ensures zero burst loss. In two-way reservation, the connection is established for each burst before transmission. The two-way reservation is not suitable for long-haul backbone optical networks due to the high RTT of the control packet but this RTT is not high in a DCN. The proposed designs employ a single-stage core topology with multiple optical switches that has the capacity to be scaled up and scaled out easily. Network-level simulation is considered to evaluate the performance of the proposed schemes.

First optical interconnection scheme proposed (HOSA) is based on the use of both fast and slow optical switches. HOSA leverages the strengths of both types of optical switch. The hybrid architecture features MEMS OXCs for low cost and uses fast

optical switches to achieve low latency. The core idea is to use fast optical switches to hide the lengthy reconfiguration procedure of the slow MEMS switches from users. HOSA features separate data and control planes. The control plane comprises a centralized controller while the data plane contains an array of fast and slow optical switches. A scalability analysis of HOSA, investigating various ratios of slow and fast optical switches, has been done. Scalability analysis shows that the proposed design is scalable to more than hundred thousand servers which is suitable for data centres of very large scale. A comparison of the cost and power consumption of the proposed design and those of conventional interconnects by using analytical modelling is also presented. The results show 50% power efficiency as compared to other conventional electrical networks while $30-35\%$ improvement in power consumption is achieved over hybrid optical/electrical network.

The trade-off between the performance and the capacity of both types of switch is also presented. The results indicate that the proposed hybrid technique, where only 40% of the interconnect capacity is provided by fast switches, shows performance comparable to that of an interconnect exclusively using fast switches till 83% load, while the cost of the hybrid architecture is reduced by 33% more than half compared to using fast switches only. However, the cost of the hybrid design is less than the Fat tree and the BCube networks.

The large aggregation time of bursts is the limitation of HOSA. In order to overcome this limitation, a second optical interconnection scheme (HOSA with TDS) is proposed. In HOSA with TDS, there is no need to aggregate large amount of traffic. The controller maintains a traffic demand matrix which updates traffic demand periodically and assigns slow paths to elephant flows. The results show low latency and high throughput for various workload communication patterns. The throughput achieved in HOSA with TDS is almost the same as that of the baseline electrical network while slightly higher latency than the baseline electrical network is achieved. However, the performance of HOSA has been improved by introducing HOSA with TDS scheme.

In HOSA with TDS, the control plane can only support applications that have high

traffic stability, i.e. workloads that last several seconds. So for dynamically changing traffic patterns, HOSA with TDS does not perform well. In the third scheme proposed, a new architecture called FOSA, which is based on using only fast optical switches, is presented. In this design, OBS with two-way reservation is used. The proposed technique shows a considerable improvement in throughput and packet loss ratio over traditional methods of OBS while delay performance comparable to traditional methods of OBS is also achieved. The proposed technique also demonstrates performance comparable to that of electrical data centre networks. FOSA is a costly design in terms of CAPEX. Its CAPEX cost is $20-30\%$ higher as compared to BCube and Fat tree networks respectively while it is almost two third to that of TE/OE networks. However, this CAPEX cost is mitigated by its greater energy efficiency.

The performance of TCP over OBS network is poor because contention is interpreted as congestion. The performance of the TCP in FOSA is investigated. The results show significant improvements in the throughput, completion time and packets loss as compared to the traditional methods of OBS. The proposed scheme also demonstrates better TCP performance than conventional electronic packet switching DCN for all types of workloads.

7.2 Future Work

To assess the performance of the proposed schemes for DCN, simulations models have been developed in the OMNeT++ simulation framework. This required the control plane algorithms to be implemented in C++ within the OMNeT++ models. Porting this code to interact with real switch hardware using, for example, appropriate extensions to the OpenFlow protocol [151], rather than with the simulation model, will allow the control plane to be deployed on generic hardware. For data plane implementation, the fast optical switches that are described in Chapter 2 can be considered. The problem of packaging the proposed solution for real-world deployment in a manner compatible with existing SDN frameworks can be addressed in future work.

The scalability analysis of the proposed architectures considers only data plane

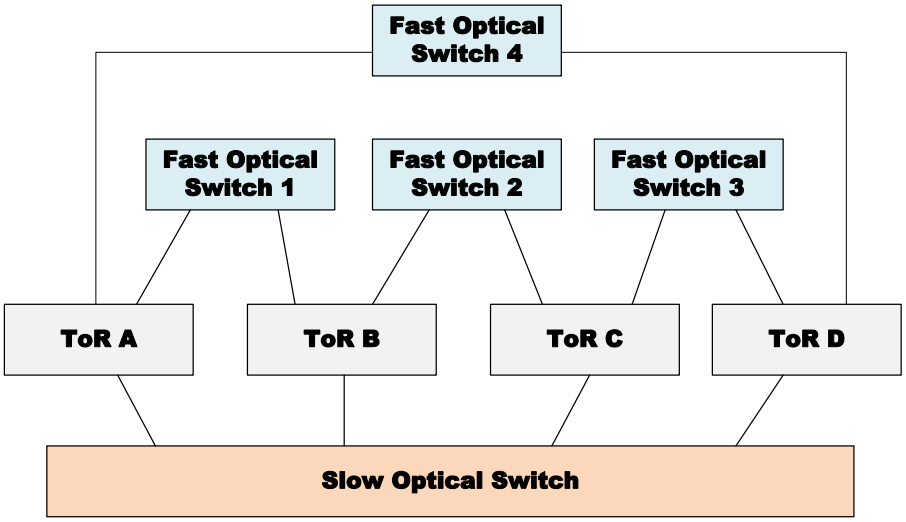


Figure 7.1. Multi-hopping technique in future work.

issues. Evaluating the performance of the control plane is an open issue as it is implementation dependent and can only be addressed after deployment of the proposed technique in a real world scenario. This may be the bottleneck in scaling up the architecture, and will be considered in future studies of the architecture. In addition to this, the feasibility of deployment of distributed control plane using multiple controllers could also be investigated.

According to the assumptions made in Section 3.2.1, this thesis considers equal size of fast and slow optical switches i.e. $N \times N$ in N rack DCN for HOSA and HOSA with TDS schemes. So, performance analysis of these two proposed techniques has been done using $N \times N$ switching fabric. This assumption is based on the literature but not on commercially available components. However, for immediate deployment of the proposed techniques, this assumption will not work because fast optical switches are not available in a large switching fabric as slow optical switches are. For example, slow optical MEMS switch are available commercially in 384 ports [56] while 128 port AWGR can be built using 128 port AWGs [63] as described in Chapter 2. So in this case, size of slow optical switch is three times larger than the size of fast optical switch. So this $N \times N$ assumption for both types of optical switches is not valid. In this case, fast optical switch will be $K \times K$ size and slow optical switch will be $N \times N$ size where $K < N$.

Two approaches can be considered as a future work to avoid the issue of unequal switching fabric in the proposed hybrid schemes. As it is described in Chapter 4 that all the racks in a DCN do not send traffic to all other racks in the DCN over a given period of time i.e. the TDC is not too high because different studies on data centre traffic [102,119,121] have shown that traffic within data centres is bounded in degrees and racks communicate with only few other racks over a given period of time. So in the first approach, idea is to physically connect those racks using fast optical switches that communicate with each other more often. This technique should work when $TDC < K$ and source ToR switch is also connected physically using fast optical switches to the destination ToR switches. However, in a scenario if ToR switch is not connected physically to other ToR switches using fast optical switches then a second approach can be considered.

In the second approach, the idea of multi-hopping can be considered similar to the one proposed in [54, 72]. In multi-hopping technique, some nodes are physically connected (i.e. single hop) while all other nodes are logically connected (i.e. multi-hop) with each other. For example, nodes A and B are physically connected and nodes B and C are physically connected while node A and C are logically connected via node B. If node A wants to send the data to the node C, it will first send the data to the node B and node B then forwards the data to the node C. The data will take multi-hops to reach at the destination. Similar approach can be used for ToR switches that are not connected physically via fast optical switches. In this way, the ToR switch will act as a bridge between two unconnected ToR switch. This scenario is shown in Figure 7.1.

Consider a scenario with network size $N = 4$ as shown in Figure 7.1. In this case, there is a slow optical switch with $N \times N$ configuration that connects all 4 ToR switches. However, fast switch is available in $K \times K$ configuration where $K = 2$. In this way, fast switch cannot connects all ToR switches. If ToR A wants to send traffic to ToR C, then there will be two options available. First, it can send through the slow switch path if it is available. Second, it can choose multi-hopping technique that will first send the traffic to ToR B which will forward the traffic to ToR C. In order to realize this scheme, there will be a need to design an efficient route selection technique that will ensure efficient performance. In future work, the feasibility of this multi-hopping technique

can be explored.

Above approaches are for proposed hybrid designs i.e. HOSA and HOSA with TDS. Since there is no slow optical switch in FOSA design, so only multi-hopping technique will be sufficient to handle all types of traffic in FOSA.

In summary, future work can be done on following points:

- Implementation of the proposed techniques and prototype development
- To investigate the scalability analysis of the control plane.
- To investigate the feasibility of multi-hopping technique for small size fast optical switches in the proposed techniques.

APPENDIX A

ToR Switch Datasheet

Aggregate/Core Switch Datasheet

MEMS Switch Datasheet

AWGR Datasheet

Data Sheet

ToR Switch Datasheet

Cisco Nexus 3064-X, 3064-T, and 3064-32T Switches

Product Overview

The Cisco Nexus® 3064-X, 3064-T, and 3064-32T Switches are high-performance, high-density Ethernet switches that are part of the Cisco Nexus 3000 Series Switches portfolio. These compact one-rack-unit (1RU) form-factor 10 Gigabit Ethernet switches provide line-rate Layer 2 and 3 switching. They run the industry-leading Cisco® NX-OS Software operating system, providing customers with comprehensive features and functions that are widely deployed globally. They support both forward and reverse airflow schemes with AC and DC power inputs. The Cisco Nexus 3064 switches are well suited for data centers that require cost-effective, power-efficient, line-rate Layer 2 and 3 top-of-rack (ToR) switches.

Three Cisco Nexus 3064 switches are available:

- Cisco Nexus 3064-X (Figure 1): This 10-Gbps Enhanced Small Form-Factor Pluggable (SFP+)-based top-of-rack switch has 48 SFP+ ports and 4 Quad SFP+ (QSFP+) ports. Each SFP+ port can operate in 100-Mbps, 1-Gbps, or 10-Gbps mode, and each QSFP+ port can operate in native 40-Gbps or 4 x 10-Gbps mode. This switch is a true phy-less switch that is optimized for low latency and low power consumption.
- Cisco Nexus 3064-T (Figure 2): This 10GBASE-T switch has 48 10GBASE-T RJ-45 ports and 4 QSFP+ ports. This switch is well suited for customers who want to reuse existing copper cabling while migrating from 1-Gbps to 10-Gbps servers.
- Cisco Nexus 3064-32T (Figure 2): This switch is the Cisco Nexus 3064-T with 32 10GBASE-T ports and 4 QSFP+ ports enabled. The ports are enabled through software licensing. This switch provides a cost-effective solution for customers who require up to 32 10GBASE-T ports per rack. This switch comes with a 32-10GBASE-T port license preinstalled. To enable the remaining 16 10GBASE-T ports, the customer installs the 16-port upgrade license.

Figure 1. Cisco Nexus 3064-X Switch

Figure 2. Cisco Nexus 3064-T and 3064-32T Switch

Main Benefits

The Cisco Nexus 3064 switches provide the following main benefits:

- Wire-rate Layer 2 and 3 switching on all ports
  - The Cisco Nexus 3064 switches provide Layer 2 and 3 switching of up to 1.2 terabits per second (Tbps) and more than 950 million packets per second (mpps) in a compact 1RU form factor.
- Ultra-low latency
  - The Cisco Nexus 3064 switches deliver ultra-low nominal latency that allows customers to implement high-performance infrastructure for high-frequency trading (HFT) workloads.
- Purpose-built on Cisco NX-OS operating system with comprehensive, proven innovations
  - Virtual PortChannel (vPC) provides Layer 2 multipathing through the elimination of Spanning Tree Protocol and enables fully utilized bisectional bandwidth and simplified Layer 2 logical topologies without the need to change the existing management and deployment models.
  - PowerOn Auto Provisioning (POAP) enables touchless bootup and configuration of the switch, drastically reducing provisioning time.
  - Cisco Embedded Event Manager (EEM) and Python scripting enable automation and remote operations in the data center.
  - Advanced buffer monitoring reports real-time buffer use per port and per queue, which allows organizations to monitor traffic bursts and application traffic patterns.
  - The 64-way equal-cost multipath (ECMP) routing enables Layer 3 fat tree designs and allows organizations to prevent network bottlenecks, increase resiliency, and add capacity with little network disruption.
  - EtherAnalyzer is a built-in packet analyzer for monitoring and troubleshooting control-plane traffic and is based on the popular Wireshark open source network protocol analyzer.
  - Precision Time Protocol (PTP; IEEE 1588) provides accurate clock synchronization and improved data correlation with network captures and system events.
  - Full Layer 3 unicast and multicast routing protocol suites are supported, including Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), Enhanced Interior Gateway Routing Protocol (EIGRP), Routing Information Protocol Version 2 (RIPv2), Protocol Independent Multicast sparse mode (PIM-SM), Source-Specific Multicast (SSM), and Multicast Source Discovery Protocol (MSDP).
- Network traffic monitoring with Cisco Nexus Data Broker
  - Build simple, scalable and cost-effective network tap or Cisco Switched Port Analyzer (SPAN) aggregation for network traffic monitoring and analysis.

Configuration

- Cisco Nexus 3064-X
  - 48 fixed 10 Gigabit Ethernet SFP+ ports (can operate at 100-Mbps, 1-Gbps, and 10-Gbps speeds)
  - Four fixed QSFP+ ports (each QSFP+ port can support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet)
- Cisco Nexus 3064-T
  - 48 fixed 10GBASE-T ports (can operate at 100-Mbps, 1-Gbps, and 10-Gbps speeds)
  - Four fixed QSFP+ ports (each QSFP+ port can support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet)

- Cisco Nexus 3064-32T
  - 32 fixed 10GBASE-T ports (can operate at 100-Mbps, 1-Gbps, and 10-Gbps speeds)
  - Four fixed QSFP+ ports (each QSFP+ port can support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet)
  - Upgrade to 48 fixed 10GBASE-T and 4 QSFP+ ports by installing a 16-port upgrade license
- Locator LED
- Dual redundant[1] power supplies
- Fan tray with redundant fans
- Two 10/100/1000-Mbps management ports
- One RS-232 serial console port
- One USB port
- Locator LED button

Support for both forward (port-side exhaust) and reversed (port-side intake) airflow schemes is available.

Transceiver and Cabling Options

The Cisco Nexus 3064 switches support a wide variety of 1, 10, and 40 Gigabit Ethernet connectivity options. 1 and 10 Gigabit Ethernet connectivity is achieved in the first 48 ports, and 40 Gigabit Ethernet connectivity is achieved using QSFP+ transceivers in the last 4 ports.

QSFP+ technology allows smooth transition from 10 to 40 Gigabit Ethernet infrastructures in data centers. The Cisco Nexus 3064 switches support connectivity over copper and fiber cables, providing excellent physical-layer flexibility. For low-cost cabling, copper-based 40-Gbps Twinax cables can be used, and for longer cable reaches, short-reach optical transceivers are excellent.

Connectivity can be established from the QSFP ports to an upstream 10 Gigabit Ethernet switch using a splitter cable that has a QSFP transceiver on one end and four SFP+ transceivers on the other end. Similar capability can be achieved using optical transceivers by procuring third-party fiber splitters. Table 1 lists the QSFP transceiver types supported.

Table 1. Cisco Nexus 3064 QSFP Transceiver Support Matrix

Part Number	Description
QSFP-4X10G-AC10M	Cisco 40GBASE-CR4 QSFP+ to 4 10GBASE-CU SFP+ direct-attach breakout cable, 10m, active
QSFP-4X10G-AC7M	Cisco 40GBASE-CR4 QSFP+ to 4 10GBASE-CU SFP+ direct-attach breakout cable, 7m, active
QSFP-4SFP10G-CU5M	QSFP to 4xSFP10G passive copper splitter cable, 5m
QSFP-4SFP10G-CU3M	QSFP to 4xSFP10G passive copper splitter cable, 3m
QSFP-4SFP10G-CU1M	QSFP to 4xSFP10G passive copper splitter cable, 1m
QSFP-H40G-ACU10M	Cisco 40GBASE-CR4 QSFP+ direct-attach copper cable, 10m, active
QSFP-H40G-ACU7M	Cisco 40GBASE-CR4 QSFP+ direct-attach copper cable, 7m, active
QSFP-H40G-CU5M	40GBASE-CR4 passive copper cable, 5m
QSFP-H40G-CU3M	40GBASE-CR4 passive copper cable, 3m
QSFP-H40G-CU1M	40GBASE-CR4 passive copper cable, 1m
QSFP-40G-SR4	40GBASE-SR4 QSFP transceiver module with MPO connector

[1] Cisco Nexus 3064-T and 3064-32T DC power supplies operate in combined mode only.

Part Number	Description
QSFP-40G-CSR4	Cisco 40GBASE-CSR4 transceiver module, MPO, 300m
QSFP-40GE-LR4	QSFP 40GBASE-LR4 QSFP+ module for SMF

For in-rack or adjacent-rack cabling, the Cisco Nexus 3064-X supports SFP+ direct-attach 10 Gigabit Ethernet copper, an innovative solution that integrates transceivers with Twinax cables into an energy-efficient and low-cost solution. For longer cable runs, multimode and single-mode optical SFP+ transceivers are supported. Table 2 lists the supported 10 Gigabit Ethernet transceiver options.

Table 2. Cisco Nexus 3064-X 10 Gigabit Ethernet Transceiver Support Matrix

Part Number	Description
SFP-10G-SR	10GBASE-SR SFP+ module (multimode fiber [MMF])
SFP-10G-LR	10GBASE-LR SFP+ module (single-mode fiber [SMF])
SFP-10G-ER	Cisco 10GBASE-ER SFP+ module for SMF
SFP-10G-ZR	Cisco 10GBASE-ZR SFP+ module for SMF*
DWDM-SFP10G-*	10GBASE-DWDM Modules (multiple varieties)
SFP-H10GB-CU1M	10GBASE-CU SFP+ cable 1m (Twinax cable)
SFP-H10GB-CU3M	10GBASE-CU SFP+ cable 3m (Twinax cable)
SFP-H10GB-CU5M	10GBASE-CU SFP+ cable 5m (Twinax cable)
SFP-H10GB-ACU7M	Active Twinax cable assembly, 7m
SFP-H10GB-ACU10M	Active Twinax cable assembly, 10m
SFP-10G-AOC1M	10GBASE-AOC SFP+ cable 1m
SFP-10G-AOC2M	10GBASE-AOC SFP+ cable 2m
SFP-10G-AOC3M	10GBASE-AOC SFP+ cable 3m
SFP-10G-AOC5M	10GBASE-AOC SFP+ cable 5m
SFP-10G-AOC7M	10GBASE-AOC SFP+ cable 7m
SFP-10G-AOC10M	10GBASE-AOC SFP+ cable 10m

The Cisco Nexus 3064-X is compatible with existing Gigabit Ethernet infrastructures. The 10 Gigabit Ethernet interfaces can operate in either Gigabit Ethernet or 100-Mbps mode. Table 3 lists the Gigabit Ethernet SFP transceivers that are supported. 100-Mbps connectivity can be achieved by using copper-based SFP transceivers (SFP-GE-T and GLC-T).

Table 3. Cisco Nexus 3064 Gigabit Ethernet Transceiver Support Matrix

Part Number	Description
SFP-GE-T	1000BASE-T NEBS 3 ESD
GLC-T	1000BASE-T SFP
GLC-SX-MM	GE SFP, LC connector SX transceiver (MMF)
GLC-LH-SM	GE SFP, LC connector LX/LH transceiver (SMF)
GLC-SX-MMD	1000BASE-SX short wavelength; with DOM
GLC-LH-SMD	1000BASE-LX/LH long-wavelength; with DOM
GLC-EX-SMD	1000BASE-EX long-wavelength; with DOM
GLC-ZX-SMD	1000BASE-ZX extended distance; with DOM
GLC-BX-U	1000BASE-BX10-U upstream bidirectional single fiber; with DOM
GLC-BX-D	1000BASE-BX10-D downstream bidirectional single fiber; with DOM

For more information about the transceiver types, see http://www.cisco.com/en/US/products/hw/modules/ps5455/prod_module_series_home.html.

The Cisco Nexus 3064-T and 3064-32T support IEEE 802.3an standard cables and transceivers to provide 10-Gbps connections over unshielded or shielded twisted-pair cables, over distances of up to 330 feet (100 meters). It provides a cost-effective and highly scalable 10 Gigabit Ethernet implementation over structured copper cabling infrastructure that is widely used in data centers.

Cisco NX-OS Software Overview

Cisco NX-OS is a data center-class operating system built with modularity, resiliency, and serviceability at its foundation. Cisco NX-OS helps ensure continuous availability and sets the standard for mission-critical data center environments. The self-healing and highly modular design of Cisco NX-OS makes zero-impact operations a reality and provides exceptional operation flexibility.

Focused on the requirements of the data center, Cisco NX-OS provides a robust and comprehensive feature set that meets the networking requirements of present and future data centers. With an XML interface and a command-line interface (CLI) like that of Cisco IOS® Software, Cisco NX-OS provides state-of-the-art implementations of relevant networking standards as well as a variety of true data center-class Cisco innovations.

Cisco NX-OS Software Benefits

Table 4 summarizes the benefits that Cisco NX-OS Software offers.

Table 4. Benefits of Cisco NX-OS Software

Feature	Benefit
Common software throughout the data center: Cisco NX-OS runs on all Cisco data center switch platforms (Cisco Nexus 7000, 5000, 4000, 2000, and 1000V Series).	• Simplification of data center operating environment • End-to-end Cisco Nexus and Cisco NX-OS fabric • No retraining necessary for data center engineering and operations teams
Software compatibility: Cisco NX-OS intercoperates with Cisco products running any variant of Cisco IOS Software and also with any networking OS that conforms to the networking standards listed as supported in this data sheet.	• Transparent operation with existing network infrastructure • Open standards • No compatibility concerns
Modular software design: Cisco NX-OS is designed to support distributed multithreaded processing. Cisco NX-OS modular processes are instantiated on demand, each in a separate protected memory space. Thus, processes are started and system resources allocated only when a feature is enabled. The modular processes are governed by a real-time preemptive scheduler that helps ensure timely processing of critical functions.	• Robust software • Fault tolerance • Increased scalability • Increased network availability
Troubleshooting and diagnostics: Cisco NX-OS is built with unique serviceability functions to allow network operators to take early action based on network trends and events, enhancing network planning and improving network operations center (NOC) and vendor response times. Cisco Smart Call Home and Cisco Online Health Management System (OHMS) are some of the features that enhance the serviceability of Cisco NX-OS.	• Quick problem isolation and resolution • Continuous system monitoring and proactive notifications • Improved productivity of operations teams
Ease of management: Cisco NX-OS provides a programmatic XML interface based on the NETCONF industry standard. The Cisco NX-OS XML interface provides a consistent API for devices. Cisco NX-OS also provides support for Simple Network Management Protocol (SNMP) Versions 1, 2, and 3 MIBs.	• Rapid development and creation of tools for enhanced management • Comprehensive SNMP MIB support for efficient remote monitoring
Using the Cisco Nexus Data Broker software and Cisco Plug-in for OpenFlow agent, the Cisco Nexus 3064 switches can be used to build a scalable, cost-effective, and programmable tap or SPAN aggregation infrastructure. This approach replaces the traditional purpose-built matrix switches with these switches. You can interconnect these switches to build a multilayer topology for tap or SPAN aggregation infrastructure.	• Scalable and cost effective • Robust traffic filtering capabilities • Traffic aggregation from multiple input ports across different switches • Traffic replication and forwarding to multiple monitoring tools

Feature	Benefit
Role-based access control (RBAC): With RBAC, Cisco NX-OS enables administrators to limit access to switch operations by assigning roles to users. Administrators can customize access and restrict it to the users who require it.	• Effective access control mechanism based on user roles • Improved network device security • Reduction in network problems arising from human error

Cisco NX-OS Software Packages for Cisco Nexus 3064 Switches

The software packages for the Cisco Nexus 3064 switches offer flexibility and comprehensive features while being consistent with the Cisco Nexus access switches. The default system software has a comprehensive Layer 2 feature set with extensive security and management features and a basic Layer 3 feature set. To enable advanced Layer 3 IP routing functions, an additional license must be installed, as described in Table 5. See Table 7 later in this document for a complete list of software features.

Table 5. Software Licensing for Cisco Nexus 3064 Switches

Software Package	Features Supported
System default: Base license (N3K-BAS1K9); included, with no additional purchase necessary)	• Comprehensive Layer 2 feature set: VLAN, IEEE 802.1Q Trunking, vPC, LACP, Unidirectional Link Detection UDLD (standard and aggressive), MSTP, RSTP, Spanning Tree guards, and Transparent VLAN Trunk Protocol (VTP) • Security: Authentication, authorization, and accounting (AAA); access control lists (ACLs), Dynamic Host Configuration Protocol (DHCP) snooping, storm control, private VLAN (PVLAN), and configurable Control-Plane Policing (CoPP) • Management features: Cisco Data Center Network Manager (DCNM) support, console, Secure Shell Version 2 (SSHv2) access, Cisco Discovery Protocol, SNMP, and syslog • Layer 3 IP routing: inter-VLAN routing (IVR), static routes, RIPv2, ACLs, OSPFv2 (limited to 256 routes), EIGRP stub, Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), and Unicast Reverse-Path Forwarding (uRPF) • Multicast: PIM SM, SSM, and MSDP
LAN Enterprise license (N3K-LAN1K9); requires Base license	• Advanced Layer 3 IP routing: OSPFv2, EIGRP, BGP, and Virtual Route Forwarding lite (VRF-lite)
Cisco Nexus Data Broker license (NDB-FX-SWT-K9)	• License for using the tap and SPAN aggregation functions with Cisco Nexus Data Broker; only the Base license is needed for this feature

Cisco Data Center Network Manager

The Cisco Nexus 3064 switches are supported in Cisco DCNM. Cisco DCNM is designed for the Cisco Nexus hardware platforms, which are enabled for Cisco NX-OS. Cisco DCNM is a Cisco management solution that increases overall data center infrastructure uptime and reliability, improving business continuity. Focused on the management requirements of the data center network, Cisco DCNM provides a robust framework and comprehensive feature set that can meet the routing, switching, and storage administration needs of present and future data centers. Cisco DCNM automates the provisioning process, proactively monitors the LAN by detecting performance degradation, secures the network, and simplifies the diagnosis of dysfunctional network elements.

Cisco Nexus Data Broker

The Cisco Nexus 3064 switches with Cisco Nexus Data Broker can be used to build a scalable and cost-effective traffic monitoring infrastructure using network taps and SPAN. This approach replaces the traditional purpose-built matrix switches with one or more OpenFlow-enabled Cisco Nexus switches. You can interconnect these switches to build a scalable tap or SPAN aggregation infrastructure. You also can combine tap and SPAN sources to bring the copy of the production traffic to this tap or SPAN aggregation infrastructure. In addition, you can distribute these sources and traffic monitoring and analysis tools across multiple Cisco Nexus switches. For more details about the Cisco Nexus Data Broker visit http://www.cisco.com/go/nexusdatabroker.

Product Specifications

Table 6 lists the specifications for the Cisco Nexus 3064 switches, Table 7 lists software features, and Table 8 lists management standards and support.

Table 6. Specifications

Description	Specification	
Physical	• 1RU fixed form factor • Cisco Nexus 3064-X: 64 10 Gigabit Ethernet ports (48 SFP+ and 4 QSFP+) ◦ 48 SFP ports support 1 and 10 Gigabit Ethernet ◦ 4 QSFP ports support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet each • Cisco Nexus 3064-T: 64 x 10 Gigabit Ethernet ports (48 10GBASE-T and 4 QSFP+) ◦ 48 RJ-45 ports support 100 Mbps and 1 and 10 Gigabit Ethernet ◦ 4 QSFP ports support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet each • Cisco Nexus 3064-32T: 48 x 10 Gigabit Ethernet ports (32 10GBASE-T and 4 QSFP+) ◦ 32 RJ-45 ports support 100 Mbps and 1 and 10 Gigabit Ethernet ◦ 4 QSFP ports support 4 x 10 Gigabit Ethernet or 40 Gigabit Ethernet each • 2 redundant power supplies • 1 fan tray with redundant fans • 1 I/O module with management, console, and USB flash memory ports	
Performance	• 1.28-Tbps switching capacity • Forwarding rate of 950 mpps • Line-rate traffic throughput (both Layer 2 and 3) on all ports • Configurable maximum transmission units (MTUs) of up to 9216 bytes (jumbo frames)	
Hardware tables and scalability	MAC addresses	128,000
	Number of VLANS	4096
	Spanning-tree instances	• Rapid Spanning Tree Protocol (RSTP): 512 • Multiple Spanning Tree (MST) Protocol: 64
	ACL entries	• 2000 ingress • 1000 egress
	Routing table	• 16,000 prefixes and 16,000 host entries* • 8000 multicast routes*
	Number of EtherChannels	64 (with vPC)
	Number of ports per EtherChannel	32
	Buffers	9 MB shared
	Boot flash memory	2 GB
Power	Number of power supplies	2 • Cisco Nexus 3064-X: Redundant for AC and DC power • Cisco 3064-T and 3064-32T: Redundant for AC power
	Power supply types	• AC (forward and reversed airflow) • DC (forward and reversed airflow)
	Typical operating power	• Cisco Nexus 3064-X ◦ 143 watts (W; 64p with Twinax at 100% load; 2 power supply units [PSUs]) ◦ 177W (64p with SR optics at 100% load; 2 PSUs) • Cisco Nexus 3064-T ◦ 362W (48p with 3m cables; 4 SR4 at 100% load)
	Maximum power	• Cisco Nexus 3064-X: 199W • Cisco Nexus 3064-T
	AC PSUs • Input voltage • Frequency	 • 100 to 240 VAC • 50 to 60 Hz

Description	Specification	
	• Efficiency	• 89 to 91% at 220V
	DC PSUs • Input voltage • Maximum current • Efficiency	 • -40 to -72 VDC • 33A • 85 to 88%
	Typical heat dissipation	• Cisco Nexus 3064-X ◦ 488 BTU/hr (64p with Twinax at 100% load; 2 PSUs) ◦ 605 BTU/hr (64p with SR optics at 100% load; 2 PSUs) • Cisco Nexus 3064-T ◦ 1235 BTU/hr (48p with 3m cables; 4 SR4 at 100% load)
	Maximum heat dissipation	• Cisco Nexus 3064-X: 683 BTU/hr • Cisco Nexus 3064-T: 1553 BTU/hr
Cooling	Forward and reversed airflow schemes: • Forward airflow: Port-side exhaust (air enters through fan-tray and power supplies and exits through ports) • Reversed airflow: Port-side intake (air enters through ports and exits through fan-tray and power supplies) Single fan tray with redundant fans Hot swappable (must swap within 1 min)	
Sound	Measured sound power (maximum) • Fan speed: 40% duty cycle • Fan speed: 60% duty cycle • Fan speed: 100% duty cycle	 • 59.7 dBA • 66.4 dBA • 71.0 dBA
Environment	Dimensions (height x width x depth)	• Cisco Nexus 3064-X: 1.72 x 17.3 x 19.7 in. (4.4 x 43.9 x 50.5 cm) • Cisco Nexus 3064-T and 3064-32T: 1.72 x 17.3 x 22.45 in.(4.4 x 43.9 x 57.0 cm)
	Weight	• Cisco Nexus 3064-X: 20.5 lb (9.3 kg) • Cisco Nexus 3064-T and 3064-32T: 20.8 lb (9.5 kg)
	Operating temperature	32 to 104°F (0 to 40°C)
	Storage temperature	-40 to 158°F (-40 to 70°C)
	Operating relative humidity	• 10 to 85% noncondensing • Up to 5 days at maximum (85%) humidity • Recommend ASHRAE data center environment
	Storage relative humidity	5 to 95% noncondensing
	Altitude	0 to 10,000 ft (0 to 3000m)

* Please refer to the Cisco Nexus 3000 Series Verified Scalability Guide for scalability numbers validated on specific software releases: http://www.cisco.com/en/US/products/ps11541/products_installation_and_configuration_guides_list.html.

Table 7. Software Features

Description	Specification
Layer 2	• Layer 2 switch ports and VLAN trunks • IEEE 802.1Q VLAN encapsulation • Support for up to 4096 VLANs • Rapid Per-VLAN Spanning Tree Plus (PVRST+) (IEEE 802.1w compatible) • Multiple Spanning Tree Protocol (MSTP) (IEEE 802.1s): 64 instances • Spanning Tree PortFast • Spanning Tree Root Guard • Spanning Tree Bridge Assurance • Cisco EtherChannel technology (up to 32 ports per EtherChannel) • Link Aggregation Control Protocol (LACP): IEEE 802.3ad • Advanced PortChannel hashing based on Layer 2, 3, and 4 information

Description	Specification
	• vPC • Jumbo frames on all ports (up to 9216 bytes) • Storm control (unicast, multicast, and broadcast) • Private VLANs
Layer 3	• Layer 3 interfaces: Routed ports on interfaces, switch virtual interfaces (SVIs), PortChannels, and subinterfaces (total: 1024) • 64-way ECMP • 2000 ingress and 1000 egress ACL entries • IPv6 routing: Static, OSPFv3, and BGPv6 • Routing protocols: Static, RIPv2, EIGRP, OSPF, and BGP • Bidirectional Flow Detection (BFD) for BGP, OSPF and ipv4 Static routes • HSRP and VRRP • ACL: Routed ACL with Layer 3 and 4 options to match ingress and egress ACLs • VRF: VRF-lite (IP VPN), VRF-aware unicast (BGP, OSPF, and RIP), and VRF-aware multicast • Unicast Reverse-Path Forwarding (uRPF) with ACL; strict and loose modes • Jumbo frame support (up to 9216 bytes) • Generic Routing Encapsulation (GRE) tunneling
Multicast	Multicast: PIMv2, PIM-SM, and SSM Bootstrap router (BSR), Auto-RP, and Static RP Multicast Source Discovery Protocol (MSDP) and Anycast RP Internet Group Management Protocol (IGMP) Versions 2 and 3
Quality of Service (QoS)	Layer 2 IEEE 802.1p (class of service [CoS]) 8 hardware queues per port Per-port QoS configuration CoS trust Port-based CoS assignment Modular QoS CLI (MQC) compliance ACL-based QoS classification (Layers 2, 3, and 4) MQC CoS marking Differentiated services code point (DSCP) marking Weighted Random Early Detection (WRED) CoS-based egress queuing Egress strict-priority queuing Egress port-based scheduling: Weighted Round-Robin (WRR) Explicit Congestion Notification (ECN) Configurable ECN (Marking) per port
Security	• Ingress ACLs (standard and extended) on Ethernet • Standard and extended Layer 3 to 4 ACLs include IPv4, Internet Control Message Protocol (ICMP), TCP, and User Datagram Protocol (UDP) • VLAN-based ACLs (VACLs) • Port-based ACLs (PACLs) • Named ACLs • ACLs on virtual terminals (vtys) • DHCP snooping with Option 82 • Port number in DHCP Option 82 • DHCP relay • Dynamic Address Resolution Protocol (ARP) inspection • Configurable CoPP
Cisco Nexus Data Broker	• Topology support for tap and SPAN aggregation • Support for QinQ to tag input source tap and SPAN ports • Traffic load balancing to multiple monitoring tools • Traffic filtering based on Layer 1 through Layer 4 header information • Traffic replication and forwarding to multiple monitoring tools • Robust RBAC • Northbound Representational State Transfer (REST) API for all programmability support

Description	Specification
Management	• POAP • Python scripting • Cisco EEM • Switch management using 10/100/1000-Mbps management or console ports • CLI-based console to provide detailed out-of-band management • In-band switch management • Locator and beacon LEDs • Configuration rollback • SSHv2 • Secure Copy (SCP) server • Telnet • AAA • AAA with RBAC • RADIUS • TACACS+ • Syslog • Syslog generation on system resources (for example, FIB tables) • Embedded packet analyzer • SNMP v1, v2, and v3 • Enhanced SNMP MIB support • XML (NETCONF) support • Remote monitoring (RMON) • Advanced Encryption Standard (AES) for management traffic • Unified username and passwords across CLI and SNMP • Microsoft Challenge Handshake Authentication Protocol (MS-CHAP) • Digital certificates for management between switch and RADIUS server • Cisco Discovery Protocol Versions 1 and 2 • RBAC • Switched Port Analyzer (SPAN) on physical layer, PortChannel, and VLAN • Tunable Buffer Allocation for SPAN • Encapsulated Remote SPAN (ERSPAN) • Ingress and egress packet counters per interface • PTP (IEEE 1588) boundary clock • Network Time Protocol (NTP) • Cisco OHMS • Comprehensive bootup diagnostic tests • Cisco Call Home • Cisco DCNM • Advanced buffer utilization monitoring • sFlow

Table 8. Management and Standards Support

Description	Specification	
MIB Support	Generic MIBs • SNMPv2-SMI • CISCO-SMI • SNMPv2-TM • SNMPv2-TC • IANA-ADDRESS-FAMILY-NUMBERS-MIB • IANAifType-MIB • IANAiprouteprotocol-MIB • HCNUM-TC • CISCO-TC • SNMPv2-MIB • SNMP-COMMUNITY-MIB	Monitoring MIBs • NOTIFICATION-LOG-MIB • CISCO-SYSLOG-EXT-MIB • CISCO-PROCESS-MIB • RMON-MIB • CISCO-RMON-CONFIG-MIB • CISCO-HC-ALARM-MIB Security MIBs • CISCO-AAA-SERVER-MIB • CISCO-AAA-SERVER-EXT-MIB • CISCO-COMMON-ROLES-MIB • CISCO-COMMON-MGMT-MIB

Description	Specification	
	• SNMP-FRAMEWORK-MIB • SNMP-NOTIFICATION-MIB • SNMP-TARGET-MIB • SNMP-USER-BASED-SM-MIB • SNMP-VIEW-BASED-ACM-MIB • CISCO-SNMP-VACM-EXT-MIB • MAU-MIB Ethernet MIBs • CISCO-VLAN-MEMBERSHIP-MIB • LLDP-MIB • IP-MULTICAST-MIB Configuration MIBs • ENTITY-MIB • IF-MIB • CISCO-ENTITY-EXT-MIB • CISCO-ENTITY-FRU-CONTROL-MIB • CISCO-ENTITY-SENSOR-MIB • CISCO-SYSTEM-MIB • CISCO-SYSTEM-EXT-MIB • CISCO-IP-IF-MIB • CISCO-IF-EXTENSION-MIB • CISCO-NTP-MIB • CISCO-VTP-MIB • CISCO-IMAGE-MIB • CISCO-IMAGE-UPGRADE-MIB	• CISCO-SECURE-SHELL-MIB Miscellaneous MIBs • CISCO-LICENSE-MGR-MIB • CISCO-FEATURE-CONTROL-MIB • CISCO-CDP-MIB • CISCO-RF-MIB Layer 3 and Routing MIBs • UDP-MIB • TCP-MIB • OSPF-MIB • BGP4-MIB • CISCO-HSRP-MIB
Standards	• IEEE 802.1D: Spanning Tree Protocol • IEEE 802.1p: CoS Prioritization • IEEE 802.1Q: VLAN Tagging • IEEE 802.1s: Multiple VLAN Instances of Spanning Tree Protocol • IEEE 802.1w: Rapid Reconfiguration of Spanning Tree Protocol • IEEE 802.3z: Gigabit Ethernet • IEEE 802.3ad: Link Aggregation Control Protocol (LACP) • IEEE 802.3ae: 10 Gigabit Ethernet (Cisco Nexus 3064-X) • IEEE 802.3ba: 40 Gigabit Ethernet • IEEE 802.3an:10GBASE-T (Cisco Nexus 3064-T and 3064-32T • IEEE 802.1ab: LLDP • IEEE 1588-2008: Precision Time Protocol (Boundary Clock)	
RFC	BGP • RFC 1997: BGP Communities Attribute • RFC 2385: Protection of BGP Sessions with the TCP MD5 Signature Option • RFC 2439: BGP Route Flap Damping • RFC 2519: A Framework for Inter-Domain Route Aggregation • RFC 2545: Use of BGPv4 Multiprotocol Extensions • RFC 2858: Multiprotocol Extensions for BGPv4 • RFC 3065: Autonomous System Confederations for BGP • RFC 3392: Capabilities Advertisement with BGPv4 • RFC 4271: BGPv4 • RFC 4273: BGPv4 MIB: Definitions of Managed Objects for BGPv4 • RFC 4456: BGP Route Reflection • RFC 4486: Subcodes for BGP Cease Notification Message • RFC 4724: Graceful Restart Mechanism for BGP • RFC 4893: BGP Support for Four-Octet AS Number Space OSPF • RFC 2328: OSPF Version 2 • 8431RFC 3101: OSPF Not-So-Stubby-Area (NSSA) Option • RFC 3137: OSPF Stub Router Advertisement	

Description	Specification
	• RFC 3509: Alternative Implementations of OSPF Area Border Routers • RFC 3623: Graceful OSPF Restart • RFC 4750: OSPF Version 2 MIB RIP • RFC 1724: RIPv2 MIB Extension • RFC 2082: RIPv2 MD5 Authentication • RFC 2453: RIP Version 2 • IP Services • RFC 768: User Datagram Protocol (UDP) • RFC 783: Trivial File Transfer Protocol (TFTP) • RFC 791: IP • RFC 792: Internet Control Message Protocol (ICMP) • RFC 793: TCP • RFC 826: ARP • RFC 854: Telnet • RFC 959: FTP • RFC 1027: Proxy ARP • RFC 1305: Network Time Protocol (NTP) Version 3 • RFC 1519: Classless Interdomain Routing (CIDR) • RFC 1542: BootP Relay • RFC 1591: Domain Name System (DNS) Client • RFC 1812: IPv4 Routers • RFC 2131: DHCP Helper • RFC 2338: VRRP IP Multicast • RFC 2236: Internet Group Management Protocol, version 2 • RFC 3376: Internet Group Management Protocol, Version 3 • RFC 3446: Anycast Rendezvous Point Mechanism Using PIM and MSDP • RFC 3569: An Overview of SSM • RFC 3618: Multicast Source Discovery Protocol (MSDP) • RFC 4601: Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised) • RFC 4607: Source-Specific Multicast for IP • RFC 4610: Anycast-RP using PIM • RFC 5132: IP Multicast MIB

Software Requirements

Cisco Nexus 3000 Series Switches are supported by Cisco NX-OS Software Release 5.0 and later. Cisco NX-OS interoperates with any networking OS, including Cisco IOS Software, that conforms to the networking standards mentioned in this data sheet.

Regulatory Standards Compliance

Table 9 summarizes regulatory standards compliance for the Cisco Nexus 3000 Series.

Table 9. Regulatory Standards Compliance: Safety and EMC

Specification	Description
Regulatory compliance	• Products should comply with CE Markings per directives 2004/108/EC and 2006/95/EC
Safety	• UL 60950-1 Second Edition • CAN/CSA-C22.2 No. 60950-1 Second Edition • EN 60950-1 Second Edition • IEC 60950-1 Second Edition • AS/NZS 60950-1 • GB4943

Specification	Description
EMC: Emissions	• 47CFR Part 15 (CFR 47) Class A • AS/NZS CISPR22 Class A • CISPR22 Class A • EN55022 Class A • ICES003 Class A • VCCI Class A • EN61000-3-2 • EN61000-3-3 • KN22 Class A • CNS13438 Class A
EMC: Immunity	• EN55024 • CISPR24 • EN300386 • KN24

Ordering Information

Table 10 provides ordering information for the Cisco Nexus 3064 switches.

Table 10. Ordering Information

Part Number	Description
Chassis	
N3K-C3064PQ-10GX	Nexus 3064-X, 48 SFP+ and 4 QSFP+ ports, with enhanced scale, low latency
N3K-C3064TQ-10GT	Nexus 3064-T, 48 10GBase-T and 4 QSFP+ ports
N3K-C3064TQ-32T	Nexus 3064-32T, 32 10GBase-T and 4 QSFP+ ports
N3K-C3064-FAN	Nexus 3064 Fan Module, Forward airflow (port side exhaust)
N3K-C3064-FAN-B	Nexus 3064 Fan Module, Reversed airflow (port side intake)
N2200-PAC-400W	N2K/3K 400W AC Power Supply, Forward airflow (port side exhaust)
N2200-PAC-400W-B	N2K/3K 400W AC Power Supply, Reversed airflow (port side intake)
NXA-PAC-500W	Nexus 3064-T 500W AC PSU, Forward airflow (port side exhaust)
NXA-PAC-500W-B	Nexus 3064-T 500W AC PSU, Reverse airflow (port side intake)
N2200-PDC-400W	N2K/3K 400W DC Power Supply, Forward airflow (port side exhaust)
N3K-PDC-350W-B	N3K Series 350W DC Power Supply, Reversed airflow (port side intake)
Software Licenses	
N3K-BAS1K9	Nexus 3000 Layer 3 Base License
N3K-LAN1K9	Nexus 3000 Layer 3 LAN Enterprise License (Requires N3K-BAS1K9 License)
NDB-FX-SWT-K9	License for Tap/SPAN aggregation using Cisco Nexus Data Broker
N3064T-32T-LIC	Factory installed 32 Port license for N3064-32T
N3064T-16T-UPG=	16 Port Upgrade License for N3064-32T
Spares	
N3K-C3064-FAN=	Nexus 3064 Fan Module, Forward airflow (port side exhaust), Spare
N3K-C3064-FAN-B=	Nexus 3064 Fan Module, Reversed airflow (port side intake), Spare
N2000-PAC-400W=	N2K/3K 400W AC Power Supply, Forward airflow (port side exhaust), Spare
N2000-PAC-400W-B=	N2K/3K 400W AC Power Supply, Reversed airflow (port side intake), Spare
NXA-PAC-500W=	Nexus 3064-T 500W AC PSU, Forward airflow (port side exhaust), Spare
NXA-PAC-500W-B=	Nexus 3064-T 500W AC PSU, Reverse airflow (port side intake), Spare
N2200-PDC-400W=	N2K/3K 400W DC Power Supply, Forward airflow (port side exhaust), Spare
N3K-PDC-350W-B=	N3K Series 350W DC Power Supply, Reversed airflow (port side intake), Spare

Part Number	Description
N3K-C3064-ACC-KIT=	Nexus 3064PQ Accessory Kit
Bundles	
N3K-C3064-X-FA-L3	Nexus 3064-X, Forward Airflow (port side exhaust), AC P/S, Base and LAN Enterprise License Bundle
N3K-C3064-X-BA-L3	Nexus 3064-X, Reversed Airflow (port side intake), AC P/S, Base and LAN Enterprise License Bundle
N3K-C3064-X-FD-L3	Nexus 3064-X, Forward Airflow (port side exhaust), DC P/S, Base and LAN Enterprise License Bundle
N3K-C3064-X-BD-L3	Nexus 3064-X, Reversed Airflow (port side intake), DC P/S, Base and LAN Enterprise License Bundle
N3K-C3064-T-FA-L3	Nexus 3064-T, Forward Airflow (port side exhaust), AC P/S, Base and LAN Enterprise License Bundle
N3K-C3064-T-BA-L3	Nexus 3064-T, Reversed Airflow (port side intake), AC P/S, Base and LAN Enterprise License Bundle
Cables and Optics	
QSFP-40G-SR4(=)	40GBASE-SR4 QSFP Transceiver Module with MPO Connector
QSFP-40G-CSR4(=)	QSFP 4x10GBASE-SR Transceiver Module, MPO, 300M
QSFP-H40G-CU1M(=)	40GBASE-CR4 Passive Copper Cable, 1m
QSFP-H40G-CU3M(=)	40GBASE-CR4 Passive Copper Cable, 3m
QSFP-H40G-CU5M(=)	40GBASE-CR4 Passive Copper Cable, 5m
QSFP-4SFP10G-CU1M(=)	QSFP to 4xSFP10G Passive Copper Splitter Cable, 1m
QSFP-4SFP10G-CU3M(=)	QSFP to 4xSFP10G Passive Copper Splitter Cable, 3m
QSFP-4SFP10G-CU5M(=)	QSFP to 4xSFP10G Passive Copper Splitter Cable, 5m
SFP-10G-SR(=)	10GBASE-SR SFP+ Module
SFP-10G-LR(=)	10GBASE-LR SFP+ Module
SFP-10G-ER(=)	Cisco 10GBASE-ER SFP+ Module for SMF
SFP-10G-ZR(=)	Cisco 10GBASE-ZR SFP+ Module for SMF
SFP-H10GB-CU1M(=)	10GBASE-CU SFP+ Cable 1 Meter
SFP-H10GB-CU3M(=)	10GBASE-CU SFP+ Cable 3 Meter
SFP-H10GB-CU5M(=)	10GBASE-CU SFP+ Cable 5 Meter
SFP-H10GB-ACU7M(=)	Active Twinax Cable Assembly, 7m
SFP-H10GB-ACU10M(=)	Active Twinax Cable Assembly, 10m
SFP-GE-T(=)	1000BASE-T NEBS 3 ESD
GLC-T(=)	1000BASE-T SFP
GLC-SX-MM(=)	GE SFP, LC Connector SX Transceiver
GLC-LH-SM(=)	GE SFP, LC Connector LX/LH Transceiver

Warranty

The Cisco Nexus 3000 Series Switches have a 1-year limited hardware warranty. The warranty includes hardware replacement with a 10-day turnaround from receipt of a return materials authorization (RMA).

Service and Support

Cisco offers a wide range of services to help accelerate your success in deploying and optimizing the Cisco Nexus 3000 Series in your data center. The innovative Cisco Services offerings are delivered through a unique combination of people, processes, tools, and partners and are focused on helping you increase operation efficiency and improve your data center network. Cisco Advanced Services uses an architecture-led approach to help you align your data center infrastructure with your business goals and achieve long-term value. Cisco SMARTnet® Service helps you resolve mission-critical problems with direct access at any time to Cisco network experts and award-winning resources. With this service, you can take advantage of the Cisco Smart Call Home service capability, which offers proactive diagnostics and real-time alerts on your Cisco Nexus 3000 Series Switches. Spanning the entire network lifecycle, Cisco Services helps increase investment protection, optimize network operations, support migration operations, and strengthen your IT expertise.

Cisco Capital Financing to Help You Achieve Your Objectives

Cisco Capital® financing can help you acquire the technology you need to achieve your objectives and stay competitive. We can help you reduce capital expenditures (CapEx), accelerate your growth, and optimize your investment dollars and ROI. Cisco Capital financing gives you flexibility in acquiring hardware, software, services, and complementary third-party equipment. And you have just one predictable payment. Cisco Capital financing is available in more than 100 countries. Learn more.

For More Information

For more information, please visit http://www.cisco.com/go/nexus3000. For information about the Cisco Nexus Data Broker, please visit http://www.cisco.com/go/nexusdatabroker.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C78-651097-10 06/16

Data Sheet

Aggregate/Core Switch Datasheet

Cisco Catalyst 4500-X Series Fixed 10 Gigabit Ethernet Aggregation Switch

Product Overview

Cisco® Catalyst® 4500-X Series Switch (Figure 1) is a fixed aggregation switch that delivers best-in-class scalability, simplified network virtualization, and integrated network services for space-constrained environments in campus networks. It meets business growth objectives with unprecedented scalability, simplifies network virtualization with support for one-to-many (Cisco Easy Virtual Networks [EVN]) and many-to-one (Virtual Switching System [VSS]) virtual networks, and enables emerging applications by integrating many network services.

The Cisco Catalyst 4500-X Series offers key innovations, including:

- **Platform Scalability:** Delivers up-to 800 Gbps of switching capacity, capable of scaling up to 1.6-Tbps capacity with the VSS technology. Future-proof investment with modular uplink and auto-detect 10 Gigabit Ethernet and 1 Gigabit Ethernet ports.
- **High Availability:** Delivers the network availability demanded by business-critical enterprise applications through comprehensive high-availability capabilities, including VSS and EVN. Furthermore, innovative features such as redundant hot swappable fans and power supplies with AC to DC, and DC to AC failover remove single point of failure in network.
- **Application Monitoring:** Enhanced application monitoring through Flexible Netflow and eight ports of line rate bidirectional Switched Port Analyzer (SPAN)/Remote Switched Port Analyzer (RSPAN). In addition Cisco IOS® XE Software provides the ability to host third-party applications.
- **Security:** Support for Cisco TrustSec™ technology as well as robust control plane policing (CoPP) to address denial of service attacks.
- **Simplified Operations:** Support for Smart Install Director, providing a single point of management enabling zero-touch deployment for new switches and stacks in campus and branch networks.

Cisco Catalyst 4500-X Series Switch Family

Cisco Catalyst 4500-X Series provides scalable, fixed-campus aggregation solutions in space-constrained environments. The solution provides flexibility to build desired port density through two versions of base switches along with optional network module, providing line-rate 10GE capability. Both the 32-port and 16-port versions can be configured with optional network modules and offer similar features. The Small Form-Factor Pluggable Plus (SFP+) interface supports both 10 Gigabit Ethernet and 1 Gigabit Ethernet ports, allowing customers to use their investment in 1 Gigabit Ethernet SFP and upgrade to 10 Gigabit Ethernet when business demands change, without having to do a comprehensive upgrade of the existing deployment. The uplink module is hot swappable.

Following are key offering from this product family:

- 32 x 10 Gigabit Ethernet Port switch with optional module slot (Figure 1)
- 16 x 10 Gigabit Ethernet Port switch with optional module slot (Figure 2)
- 8 x 10 Gigabit Ethernet Port uplink module (Figure 3)

Figure 1. 32 x 10 Gigabit Ethernet Port Switch with Optional Uplink Module Slot

Figure 2. 16 x 10 Gigabit Ethernet Port Switch with Optional Uplink Module Slot

Figure 3. 8 x 10 Gigabit Ethernet Port Uplink Module

In addition, both 32 port and 16 port versions are available with front-to-back and back-to-front airflow. The front-to-back airflow switch comes with matching burgundy color fan and power supply handle to indicate warm side. Similarly, back-to-front airflow switch fan and power supply handles are color-coded in blue to indicate cool side. Figure 5 and Figure 6 show rear view of the switch with front-to-back and back-to-back airflow respectively.

Figure 4. Front-to-Back Airflow Rear View

Figure 5. Back-to-Front Airflow Rear View

Cisco Catalyst 4500-X switch provides redundant hot swappable fans and power supplies (Figure 7) for highest resiliency with no single point of failure.

Figure 6. Redundant Fan and Power Supply

Cisco Catalyst 4500-X Switch Series Feature Highlights

Cisco Catalyst 4500-X Series Switch provides nonblocking 10 Gigabit Ethernet per port bandwidth and Cisco IOS Flexible NetFlow for optimized application visibility. In addition to this, the enterprise-class Cisco Catalyst 4500-X offers the following:

- **Performance and scalability**
  - 800-Gbps switching capacity with up to 250 Mpps of throughput
  - External USB and SD card support for flexible storage options
  - 10/100/1000 RJ-45 console and management port
  - IPv6 support in hardware, providing wired-network-rate forwarding for IPv6 networks and support for dual stack with innovative resource utilization
  - Dynamic hardware forwarding-table allocations for ease of IPv4-to-IPv6 migration
  - Scalable routing (IPv4, IPv6, and multicast) tables, Layer 2 tables, and ACL and quality of service (QoS) entries to make use of eight queues per port and comprehensive security policies per port

- **Infrastructure services**
  - Cisco IOS XE Software, the modular open application platform for virtualized borderless services
  - Maximum resiliency with redundant components, Nonstop Forwarding/Stateful Switchover (NSF/SSO), and In-Service Software Upgrade (ISSU) support in a VSS enabled system
  - Network virtualization through Multi-VRF technology for Layer 3 segmentation
  - Automation through Embedded Event Manager (EEM), Cisco Smart Call Home, AutoQoS, and Auto SmartPorts for fast provisioning, diagnosis, and reporting
- **Cisco Borderless Networks services**
  - Optimized application performance through deep visibility with Flexible NetFlow supporting rich Layer 2/3/4 information (MAC, VLAN, TCP flags) and synthetic traffic monitoring with IP service-level agreement (SLA)
  - Medianet capabilities to simplify video quality of service, monitoring, and security. In addition, multicast features such as Protocol Independent Multicast (PIM) and Source-Specific Multicast (SSM) provide enterprise customers with the additional scalability to support multimedia applications
- **Investment protection and reduced TCO**

  Cisco Catalyst 4500-X Series eliminates the need for standalone solutions by integrating many network services. Customers can lower the total cost of ownership while streamlining management and accelerating deployment time. Integrated network services available on Cisco Catalyst 4500-X Series include:
  - Application visibility and control (Flexible NetFlow, Cisco IOS Embedded Event Manager)
  - Security with Cisco TrustSec[1]
  - Troubleshooting video or any User Datagram Protocol-based flows (Mediatrace)
  - Video network readiness assessment (built-in traffic simulator with IP SLA Video Operation)
  - Ability to run third-party applications (Wireshark)

Table 1 highlights the performance and scalability enhancements of the Cisco Catalyst 4500-X Series Switches.

Table 1. Cisco Catalyst 4500-X Switch Series Performance and Scalability Features

Product Number	Description
System	
Base System	Front to Back Airflow: • 32x10 GE SFP+/SFP - WS-C4500X-32SFP+ • 16x10 GE SFP+/SFP - WS-C4500X-16SFP+ Back to Front Airflow: • 32x10 GE SFP+/SFP - WS-C4500X-F-32SFP+ • 16x10 GE SFP+/SFP - WS-C4500X-F-16SFP+
Expansion Module (Optional)	8x10 GE SFP+/SFP - C4KX-NM-8SFP+
Management Port	10/100/1000 Base-T
USB Port	Type A (storage and boot) up-to 4 GB
Dual Power Supply	Yes
Field Replaceable Fans	Yes (5 fans)
Fan Redundancy	No performance impact with single fan failure
Scalability	
System Throughput	Up to 800 Gbps
IPv4 Routing in Hardware	Up to 250 Mpps

Product Number	Description
IPv6 Routing in Hardware	Up to 125 Mpps
L2 Bridging in Hardware	Up to 250 Mpps
Media Access Control (MAC) Entries	55K
Forwarding Entries	32x10 GE Port Base SKU: IPv4: 256K, IPv6: 128K 16x10 GE Port Base SKU: IPv4: 64K, IPv6: 32K
Flexible Netflow Entries	128K
Switched Port Analyzer (SPAN), Remote Switched Port Analyzer (RSPAN)	8 line rate bidirectional sessions (ingress and egress)
Total VLANs	4094
Total Switched Virtual Interfaces (SVIs)	4094
IGMP groups	32K
Multicast routes	32x10 GE Port Base SKU: IPv4: 32K, IPv6: 32K 16x10 GE Port Base SKU: IPv4: 24K, IPv6: 12K
Dynamic Host Configuration Protocol (DHCP) Snooping Entries	12K (DHCP snooping bindings)
ARP Entries	47K
Spanning Tree Protocol Instances	10K
Jumbo Frame Support for Bridged and Routed Packets	Up to 9216 bytes
High Availability and Resiliency	
High Availability Solution	Virtual Switching System (VSS)
Number of stackable switches in VSS	Up to 2
VSS Throughput	Up to 1.6 Tbps
Virtual Switch Link	1GE or 10GE
Max number of Virtual Switch Links	8
In-Service Software Upgrade	Across the switches
Nonstop Forwarding with Stateful Switchover	Across the switches
CPU and Memory	
Onboard Memory (SRAM DDR-II)	4 GB
Port Buffers	32-MB Shared Memory
CPU	Dual Core 1.5 GHz
NVRAM	2 GB
Optional External Memory (SD Card)	2 GB
QoS Features	
Port Queues	8 Queues/Port
CPU Queues	64
QoS Entries	128K (64K ingress and 64K in egress) Shared with ACL
Aggregate Rate-Limiting	Ingress port or VLAN and egress VLAN or Layer 3 port
Rate-Limiting Level Types	Committed Information Rate (CIR), Peak Information Rate (PIR)
Aggregate Traffic Rate-Limiting Policers (1K=1024)	16K

Product Number	Description
Flow-Based Rate-Limiting Method; Number of Rates	Supported using flow-records in the classification criteria and policing action
Qos Policy Enforcement	Per Port or Per Vlan or Per Port, Per VLAN Granularity
Class of Service (CoS)	Yes
Differentiated Services Code Point (DSCP)	Yes
Security Features	
Port Security	Yes
IEEE 802.1x and 802.1x Extensions	Yes
VLAN, Router, and Port ACLs	Yes
Security ACL Entries (1K=1024)	128K (64K ingress and 64K in egress) Shared with QoS
Unicast Reverse Path Forwarding (uRPF) Check in Hardware	Yes
CPU Rate Limiters (DoS Protection) Includes Control Plane Policing	Yes
Private VLANs	Yes
Micro Flow Policer	Yes. Supported using flow records in the class-map
CPU HW Rate Limiters by Packet Per Second (pps) and Bit Rate Policers (bps)	Supported in hardware control-plane policing (CoPP)
Control Plane Policing (CoPP) for Multicast	Yes
ACL Labels	Yes
Port ACL	Yes
Traffic Storm Control (formally known as Broadcast/Multicast Suppression)	Yes
Virtualization Features	
VRF-Lite Scalability	64
Easy Virtual Network (EVN) Scalability	32
Simplified Operations	
Smart Install	Smart Install Director[2]

[2] Smart Install Director support in VSS mode will be available in a future software release.

Continued Innovations Through Infrastructure Services

Modular Open Application Platform, Cisco IOS XE Software

Cisco IOS XE Software is the open service platform software operating system for the Cisco Catalyst 4500-X Series. Cisco continues to evolve Cisco IOS Software to support next-generation switching hardware and provide increased architectural flexibility to deliver Cisco Borderless Networks services. Cisco IOS XE Software provides the following customer benefits:

- Cisco IOS XE Software provides an enhanced operating system that can take advantage of the multicore CPU architecture of the Cisco Catalyst 4500-X system.
- Cisco IOS XE enables single software image, without the need to download a separate software image per license feature set.

- Cisco IOS XE Software provides customer investment protection in the existing Cisco IOS Software by keeping a consistent feature set and operational look and feel. This supports a transparent migration experience.
- Cisco IOS XE Software supports service virtualization capability that allows the Cisco Catalyst 4500-X to host third-party applications in parallel with Cisco IOS Software. The hosted application communicates with Cisco IOS Software to use its rich feature sets. This benefit keeps Cisco IOS Software simple and robust while allowing the customer to quickly adopt new technologies using proven code. Cisco IOS XE Software enables Cisco Catalyst 4500-X to be an open service platform and is a primary anchor for future Cisco Borderless Networks innovations.

Simplified Operations Through Automation

As campus switching has grown to support increasing enterprise demands, so has the need to deploy and manage new and evolving technologies. Simplified operations are critical in meeting these challenges and achieving increased operational efficiency through proactive management and reduction in unplanned network downtime. The Cisco Catalyst 4500-X offers the following rich set of capabilities for simplified operations:

- Auto Install and AutoQoS for fast deployment
- Smart Install Director support for plug-and-play configuration and image-management
- Flexible NetFlow and IP SLA for enhanced visibility
- EEM integration with NetFlow and third-party applications
- Smart Call Home, Generic Online Diagnostic (GOLD), and Digital Optical Monitoring (DOM) for simplified operations
- Cisco EnergyWise for simplified and effective power management
- ISSU, SSO, and NSF for simplified change management and high availability for VSS enabled deployment
- Configuration rollback for improved configuration management

Best-in-Class Resiliency

The Cisco Catalyst 4500-X Series is designed for excellent nonstop communications with non-interrupted hardware switching. With Cisco IOS XE Software, customers continue to reap the benefits of this best-in-class resiliency in various ways.

In addition to redundant power supplies and fans, the Cisco Catalyst 4500-X is Virtual Switching System (VSS).

Any two Cisco Catalyst 4500-X Series Switches can be pooled together into a VSS. The two switches are connected with 10 Gigabit Ethernet links called Virtual Switch Links (VSLs). Once a VSS is created, it acts as a single virtual Cisco Catalyst switch delivering the following benefits:

Operational Manageability

- Two Cisco Catalyst 4500-X Series Switches share a single point of management, single gateway IP address, and single routing instance.
- Eliminates the dependence on First Hop Redundancy Protocols (FHRP) and Spanning Tree Protocol.

Scales to 1.6 Tbps

- Scales system bandwidth capacity to 1.6 Tbps by activating all available bandwidth across redundant Cisco Catalyst 4500-X Series Switches.
- Provides up to 80 ports of 10 Gigabit Ethernet per system.

Enhanced Application Visibility with Flexible NetFlow

Cisco IOS Flexible NetFlow is the next generation in flow monitoring technology, allowing optimization of the network infrastructure resources, reducing operation costs, and improving capacity planning and security incident detection with increased flexibility and scalability. The Cisco Catalyst 4500-X Series provides 128K Flexible NetFlow entries. Based on a custom-built ASIC, Cisco Catalyst 4500-X Series delivers unprecedented flexibility and comprehensive flow visibility extending from Layer 2 (MAC, VLAN) to Layer 4 (TCP, UDP flags, and so on).

The flow data collected by Flexible NetFlow can be exported to an external collector for analysis and reporting or tracked by EEM. The Cisco Catalyst 4500-X Series enables powerful on-box and customizable event correlation and policy actions with EEM. This allows the switches to trigger customized event alarms or policy actions when the predefined condition is met. With no external appliance required, customers are able to use existing infrastructure to perform traffic monitoring, making traffic analysis economical even on large IP networks.

Additional details on Cisco Flexible NetFlow are available at: http://www.cisco.com/go/fnf.

Features at a Glance

- **Cisco IOS XE Software IP Base:** Includes all Layer 2 features and some basic Layer 3 features.
- **Cisco IOS XE Software Enterprise Services:** Upgradable with a Software Activation License (SAL); supports full Layer 3 protocols and advanced features such as complete routing scalability, Border Gateway Protocol (BGP), Virtual Routing and Forwarding, Policy-Based Routing, and so on.

These features can be enabled using the software-licensing mechanism. For details about software licensing, see “Licensing” section later in this document or visit http://www.cisco.com/go/sa.

Industry Standards

- Ethernet: IEEE 802.3
- 10 Gigabit Ethernet: IEEE 802.3ae
- IEEE 802.1D Spanning Tree Protocol
- IEEE 802.1w Rapid Reconfiguration of Spanning Tree
- IEEE 802.1s Multiple VLAN Instances of Spanning Tree
- IEEE 802.3ad LACP
- IEEE 802.1p CoS Prioritization
- IEEE 802.1Q VLAN
- IEEE 802.1X User Authentication
- IEEE 802.1x-Rev
- RMON I and II standards
- USGv6 and IPv6 Gold Logo certified

Supported Pluggables

For details about the different optical modules and the minimum Cisco IOS Software release required for each of the supported optical modules, visit: http://www.cisco.com/en/US/products/hw/modules/ps5455/products_device_support_tables_list.html.

Note: SFP-10G-ZR modules are not supported on ports 1 to 32 (or 1 to 16) in the back-to-front airflow configuration. They are supported on the uplink module ports instead. In the back-to-front airflow configuration, limit usage of ZR optics to the uplink module only.

Software Requirements

The Cisco Catalyst 4500-X Series is supported in Cisco IOS Software with minimum Cisco IOS XE Software Release 3.3.0SG. For VSS capability, minimum software requirement is Cisco IOS XE Software Release 3.4.0SG.

Environmental Conditions

Table 2 lists environmental conditions for Cisco Catalyst 4500-X Series.

Table 2. Environmental Conditions for the Cisco Catalyst 4500-X Series

Parameter	Performance Range
Operating Temperature	0°C to 40°C (RH to 90%)
Storage Temperature	-40°C to 70°C (RH 93%)
Operating Altitude	60m below sea level to 3000m above sea level
Relative Humidity	Nonoperating Humidity: 95% RH
Acoustic Noise Measured per ISO 7779 and Declared per ISO 9296 Bystander Positions Operating to an Ambient Temperature of 25°C	Industrial Product: 65 dBA maximum
RoHS	Reduction of Hazardous Substances (ROHS) 5

Power Information

Table 3 lists power information for Cisco Catalyst 4500-X Series.

Table 3. Power Supply Information for Cisco Catalyst 4500-X Series

Power Supply Feature	Support in the 4500-X Series
AC Power Max Rating	750W
System Power Consumption	330W nominal/400W max
Input-Voltage Range and Frequency	AC 100 to 240 VAC 50-60 Hz/DC -72 VDC to -40 VDC
DC Power Max Rating	750W
AC to DC failover and vice versa	Yes
Total Output BTU (Note: 1000 BTU/hr = 293W)	1122 BTU/hr (330 W) nominal/1365 BTU/hr (400 W) max
Input Current	AC 11A @ 110VAC, 6 A @ 200VAC/DC 25A Max
Output Ratings	12V @ 62A & 3.3V @ 3A
Output Holdup Time	AC = 16 ms; DC = 4 ms @ maximum load
Power-Supply Input Receptacles	AC IEC 60320 C15/DC Custom detachable screw terminal (supplied)
Power Cord Rating	AC 15A/DC 25A

MTBF Information

Table 4 lists mean-time-between-failures (MTBF) information for Cisco Catalyst 4500-X Series.

Table 4. MTBF Information for Cisco Catalyst 4500-X Series

Product Number	Description
WS-C4500X-16SFP+	209,330
WS-C4500X-24X-ES	209,330
WS-C4500X-32SFP+	199,720
WS-C4500X-40X-ES	199,720
C4KX-NM-8SFP+	2,286,500
WS-C4500X-F-16SFP+	209,330
WS-C4500X-F-32SFP+	199,720
C4KX-FAN-F	L10 Life 60,000 at 40C[1]
C4KX-FAN-R	L10 Life 60,000 at 40C
C4KX-PWR-750AC-F	1,045,265
C4KX-PWR-750AC-R	1,045,265
C4KX-PWR-750DC-F	443,423
C4KX-PWR-750DC-R	443,423

[1] Since fan is an electro-mechanical device it doesn't follow electronics failure mode. L10 life means the time 10% of total PS population will fail at a particular temperature.

Regulatory Standards Compliance

Table 5 shows regulatory standards compliance information, and Table 6 provides ordering information.

Table 5. Cisco Catalyst 4500-X Regulatory Standards Compliance

Standard	Specification
Regulatory Compliance	CE marking
EMI and EMC Compliance	47CFR Part 15 (CFR 47) Class A AS/NZS CISPR22 Class A CISPR22 Class A EN55022 Class A ICES003 Class A VCCI Class A EN61000-3-2 EN61000-3-3 KN22 Class A CNS13438 Class A EN55024 CISPR24 EN300386 KN24
Safety Certifications	UL 60950-1 Second Edition CAN/CSA-C22.2 No. 60950-1 Second Edition EN 60950-1 Second Edition IEC 60950-1 Second Edition AS/NZS 60950-1
Industry EMC, Safety, and Environmental Standards	GR-63-Core Network Equipment Building Systems (NEBS) Level 3 GR-1089-Core Level 3

Table 6. Ordering Information

Product Number	Description
Base Switch PIDs	
WS-C4500X-16SFP+	Catalyst 4500-X 16 Port 10GE IP Base, Front-to-Back Cooling, No P/S
WS-C4500X-24X-IPB	Catalyst 4500-X 24 Port 10GE IP Base, Front-to-Back Cooling, No P/S
WS-C4500X-24X-ES	Catalyst 4500-X 24 Port 10GE Enterprise Services, Front-to-Back Cooling, No P/S
WS-C4500X-32SFP+	Catalyst 4500-X 32 Port 10GE IP Base, Front-to-Back Cooling, No P/S
WS-C4500X-40X-ES	Catalyst 4500-X 40 Port 10GE Enterprise Services, Front-to-Back Cooling, No P/S
C4KX-NM-8SFP+	Catalyst 4500-X 8 Port 10GE Network Module
WS-C4500X-F-16SFP+	Catalyst 4500-X 16 Port 10GE IP Base, Back-to-Front Cooling, No P/S
WS-C4500X-F-32SFP+	Catalyst 4500-X 32 Port 10GE IP Base, Back-to-Front Cooling, No P/S
FRU and OIR FANs	
C4KX-FAN-F	Catalyst 4500-X Back-to-Front Cooling Fan
C4KX-FAN-R	Catalyst 4500-X Front-to-Back Cooling Fan
Power Supply	
C4KX-PWR-750AC-F	Catalyst 4500-X 750W AC Back-to-Front Cooling Power Supply
C4KX-PWR-750AC-R	Catalyst 4500-X 750W AC Front-to-Back Cooling Power Supply
C4KX-PWR-750DC-F	Catalyst 4500-X 750W DC Back-to-Front Cooling Power Supply
C4KX-PWR-750DC-R	Catalyst 4500-X 750W DC Front-to-Back Cooling Power Supply
Accessories	
CAB-CON-C4K-RJ45	Console Cable 6ft with RJ-45-to-RJ-45
SD-X45-2GB-E	Cisco Catalyst 4500 2-GB SD card
USB-X45-4GB-E	Cisco Catalyst 4500 4-GB USB device
Software	
S45XU-33-1511SG	Cisco IOS Software XE Release 3.3.0 SG non-crypto universal image for Cisco Catalyst 4500-X 32-port and 40-port models
S45XUK9-33-1511SG	Cisco IOS Software XE Release 3.3.0 SG crypto universal image for Cisco Catalyst 4500-X 32-port and 40-port models
S45XU-331-1511SG	Cisco IOS Software XE Release 3.3.1 SG non-crypto universal image for Cisco Catalyst 4500-X 16-port and 24-port models
S45XUK9-331-1511SG	Cisco IOS Software XE Release 3.3.1 SG crypto universal image for Cisco Catalyst 4500-X 16-port and 24-port models
S45XU-34-1512SG	Cisco IOS Software XE Release 3.4.0 SG non-crypto universal image for all Cisco Catalyst 4500-X models
S45XUK9-34-1512SG	Cisco IOS Software XE Release 3.4.0 SG crypto universal image for all Cisco Catalyst 4500-X models
C4500X-LIC=	Base product ID for software upgrade licenses on Catalyst 4500-X (paper delivery)
C4500X-IPB	Catalyst 4500-X IP BASE software license (paper delivery)
C4500X-16P-IP-ES	Catalyst 4500-X IP BASE to Enterprise Services upgrade license (paper delivery) for 16-port and 24-port models
C4500X-IP-ES	Catalyst 4500-X IP BASE to Enterprise Services upgrade license (paper delivery) for 32-port and 40-port models
L-C4500X-LIC=	Catalyst 4500-X Base product ID for software upgrade licenses (electronic delivery)
L-C4500X-IPB	Catalyst 4500-X IP BASE software license (electronic delivery)
L-C4500X-16P-IP-ES	Catalyst 4500-X IP BASE to Enterprise Services upgrade license (electronic delivery) for 16-port and 24-port models
L-C4500X-IP-ES	Catalyst 4500-X IP BASE to Enterprise Services upgrade license (electronic delivery) for 32-port and 40-port models

Licensing

Software Activation Licensing

The Cisco Catalyst 4500-X Series enables software activation licensing. Each Cisco Catalyst 4500-X Series ships with a universal image containing all feature sets, IP Base and Enterprise Services. The level of functionality is determined by the license applied.

The software activation licensing enables customers to:

- Speed deployment and roll out new Cisco software activation feature sets across global networks
- Centrally and more accurately manage and track software and license compliance
- Easily conduct software compliance audits to meet regulations without affecting network operations

Additional benefits of Cisco activation licensing include:

- Operational simplicity
  - Simplified upgrades and license transfers save time and improve productivity. You can add new capabilities simply by using a license file.
  - You can easily track software assets, licenses, and feature set status.
  - A single software image improves service delivery.
- Ease of ordering:
  - “Try and buy” lets you use a temporary license to try and evaluate new Cisco IOS Software functionality before purchasing.
  - Pay-as-you-grow software key enables new features incrementally without service calls.

For more information about Cisco software licensing, visit: http://www.cisco.com/go/sa.

Cisco ONE Software

Cisco ONE Software for Access Switching is available for the Cisco Catalyst 4500-X Series Switches.

Cisco ONE Software is a new way for customers to purchase and use our infrastructure software. It offers a simplified consumption model, centered on common customer scenarios in the data center, WANs, and LANs.

Cisco ONE Software and services provide customers with four primary benefits:

- Software suites that address typical customer use scenarios at an attractive price
- Investment protection of their software purchase through software services-enabled license portability
- Access to ongoing innovation and new technology with Cisco Software Support Service (SWSS)
- Flexible licensing models to smoothly distribute customer's software spend over time

For ordering information for Cisco ONE Software for the Cisco Catalyst 4500-X Series Switches, go to http://www.cisco.com/c/en/us/products/software/one-access/switching-part-numbers.html.

Cisco Limited Lifetime Hardware Warranty

The Cisco limited lifetime hardware warranty (LLW) includes 10-day advance hardware replacement for as long as the original end user owns the product. Table 7 describes the Cisco limited lifetime hardware warranty.

The formal warranty statement, including the warranty applicable to Cisco software, appears in the Cisco information packet that accompanies your Cisco product. We encourage you to review carefully the warranty statement shipped with your specific product before use.

For additional information on warranty terms, visit: http://www.cisco.com/go/warranty.

Table 7. Cisco Limited Lifetime Hardware Warranty

Warranty Terms	Description[1]
Warranty Duration	As long as the original end user continues to own or use the product.
EoL Policy	In the event of discontinuance of product manufacture, Cisco warranty support is limited to 5 years from the announcement of discontinuance.
Hardware Replacement	Cisco or its service center will use commercially reasonable efforts to ship a replacement part within 10 business days after receipt of the RMA request and confirmation that a replacement part is the appropriate response. Actual delivery times may vary depending on customer location.
Effective Date	Hardware warranty commences from the date of shipment to the customer (and in case of resale by a Cisco reseller, not more than 90 days after original shipment by Cisco).
Cisco Technical Assistance Center (TAC) Support	None.
Cisco.com Access	Warranty allows guest access only to Cisco.com.

[1] Cisco reserves the right to refund the purchase price as its exclusive warranty remedy.

Adding a Cisco Technical Services contract to your device coverage provides benefits not available through the warranty, including access to the Cisco Technical Assistance Center (TAC), a variety of hardware replacement options to meet critical business needs, updates for licensed Cisco IOS Software, and registered access to the extensive Cisco.com knowledge base and support tools. Choose from a flexible suite of support services designed to meet your business needs and help you maintain high-quality network performance while controlling operational costs. Table 8 describes the benefits and features of Cisco Technical Services. For more information about Cisco Technical Services, visit: http://www.cisco.com/go/ts.

Table 8. Cisco Technical Services for Cisco Catalyst 4500-X Series Switches

Technical Services
Cisco SMARTnet™ Service • Around-the-clock, global access to the Cisco TAC • Unrestricted access to the extensive Cisco.com resources, communities, and tools • Next-business-day, 8x5x4, 24x7x4, and 24x7x2 advance hardware replacement[2] and onsite parts replacement and installation available • Ongoing operating system software updates within the licensed feature set[1] • Proactive diagnostics and real-time alerts on Smart Call Home-enabled devices
Cisco Smart Foundation Service • Next-business day advance hardware replacement as available • Business hours access to small and medium-sized business (SMB) TAC (access levels vary by region) • Access to Cisco.com SMB knowledge base • Online technical resources through Cisco Smart Foundation Portal • Operating system software bug fixes and patches

Technical Services
Cisco Focused Technical Support Services Three levels of premium, high-touch services are available: • Cisco High-Touch Operations Management Service • Cisco High-Touch Technical Support Service • Cisco High-Touch Engineering Service Valid Cisco SMARTnet Service or service provider base contracts on all network equipment are required.

Footnotes:

[1] Cisco operating system updates include the following: maintenance releases, minor updates, and major updates within the licensed feature set.

[2] Advance hardware replacement is available in various service-level combinations. For example, 8x5xNBD indicates that shipment will be initiated during the standard 8-hour business day, 5 days a week (the generally accepted business days within the relevant region), with next business day (NBD) delivery. Where NBD is not available, same day ship is provided. Restrictions apply; please review the appropriate service descriptions for details.

Cisco and Partner Services

Enable the innovative, secure, intelligent edge in Cisco Borderless Network Architecture using personalized services from Cisco and our partners. Through a discovery process that begins with understanding your business objectives, we help you integrate the next-generation Cisco Catalyst 4500-X Series Switches into your architecture and incorporate network services onto that platform. Sharing knowledge and leading practices, we support your success every step of the way as you deploy, absorb, manage, and scale new technology.

For additional information about Cisco services, visit: http://www.cisco.com/go/services.

Cisco Capital

Financing to Help You Achieve Your Objectives

Cisco Capital can help you acquire the technology you need to achieve your objectives and stay competitive. We can help you reduce CapEx. Accelerate your growth. Optimize your investment dollars and ROI. Cisco Capital financing gives you flexibility in acquiring hardware, software, services, and complementary third-party equipment. And there's just one predictable payment. Cisco Capital is available in more than 100 countries. Learn more.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C78-696791-11 01/16

MEMS Switch Datasheet

Features and Benefits

The explosion of video and other Internet data is driving the demand for flexible, scalable, high-bandwidth networks. CALIENT's S320 Optical Circuit Switch is a reliable and cost-effective solution for these networks because the technology is transparent to data speed, and is protocol agnostic, thus it offers very high bandwidth and configuration flexibility as networks grow in speed from 10Gbps to 40Gbps, 100Gbps, and beyond.

Based on field proven 3D Optical MEMS technology that CALIENT has deployed in more than 100,000 optical connections globally, the new S320 Optical Circuit Switch delivers a sweet-spot of high reliability, small form factor, low power consumption and cost, and ease of use that allows the benefits of true all-optical switching to be realized in a wide range of data center and service provider applications.

Applications

The S320 provides the scalable and protocol independent automated optical switching and management infrastructure for a wide range of Data Center, Service Provider, and Government applications including:

- Flexible, scalable on-demand optical layer optimization in enterprise and cloud computing data centers and metro software defined networks (SDN)
- Rapid disaster-recovery from multiple network failure scenarios in any optical network application
- Remote configuration and restoration of high-value subsea cable networks
- High port count colorless, directionless and contention-less (CDC) ROADMs in fiber-optic service provider networks
- Fiber To The Home (FTTH/FTTP) network automation - automated service activation & testing
- Cyber security: Protection of critical network infrastructure from cyber attacks
- Sharing of high-value testing resources in lab automation & Cyber-range applications
- Software defined optical switching in multi-tenant data centers and co-location facilities

AT A GLANCE

- Small Size: 320 Ports (Tx/Rx pairs) in 7RU Chassis (LC Connectors)
- Low Power Operation: 45 Watts typical
- Low Cost: Supports deployment in data center, service provider, and government networks
- Ultra-low Latency: All-optical connectivity adds negligible latency.
- Scalable: Supports all data rates to 100 Gbps and beyond
- Reliable: Based on proven 3D MEMS design deployed in over 100,000 fiber terminations globally
- Simple to install, integrate and use: GUI-driven, EMS-ready, supports TL1, SNMP, CORBA, and OpenFlow
- Low loss: 3.5 dB maximum insertion loss
- Built-in power monitoring: Every in/out fiber is monitored providing powerful network diagnostic capabilities

PHOENIX DATACOM
NETWORK PERFORMANCE AND SECURITY

CALIENT 3D MEMS

Move the Light, not the Fiber™

Description

The S320 Optical Circuit Switch is a 320 port all-optical (OOO) switch that establishes, monitors and changes connections between single-mode optical fibers using Micro-Electro-Mechanical Systems (MEMS) optical switching. Connections are made between fibers carrying signals with any data rate or protocol. Any input fiber on the S320 can be connected to any output fiber.

The core of the S320 Optical Circuit Switch is the MEMS Switch Module (MSM). Input fibers are connected to the MSM, which establishes connections with any of the desired output fibers. Light tapped from the input and output fibers is fed to the Optical Monitoring Module (OMM) to enable monitoring of existing connections and establishment and optimization of new connections. Light is tapped using fiber tap couplers and mirror drivers control each connection by supplying voltage to each MEMS mirror.

Light is directed from the input fibers to the output fibers using arrays of tiny silicon mirrors that are fabricated using the proven CALIENT MEMS process. An optical signal transmitted through the S320 passes through three sections of the MSM320: the input collimator array, which directs the light from each input fiber to its input mirror; the mirror matrix, an array of MEMS input mirrors and an array of MEMS output mirrors; and the output collimator array, which couples light from each output mirror back into its output fiber. High-quality mirrors and collimators and precise electrostatic control of the position of each mirror, enable typical switch times of less than 50 ms and optical loss that is less than 3.5 dB for the S320 Optical Circuit Switch.

Users manage and communicate with the S320 Optical Circuit Switch via high-reliability redundant Control Processors. TL1 command sets and SNMPv3 are supported in addition to a CORBA interface and a Web-based Graphical User Interface. An OpenFlow API for SDN applications is also available.

ABOUT CALIENT

CALIENT Technologies is the global leader in Optical Circuit Switching with systems that enable dynamic optical layer optimization in next generation datacenters and software-defined networks. CALIENT's 3D MEMS switches have demonstrated years of reliability, and with more than 100,000 optical terminations shipped, CALIENT has one of the largest installed bases of photonic switches worldwide.

For more information please contact
Phoenix Datacom
Tel: 01296 397711
Email: info@phoenixdatacom.com
Web: www.phoenixdatacom.com

Specifications

OPTICAL
320 Ports In, 320 Ports Out (Each port is TX/RX pair)
Single-mode fiber, Wavelength Range: 1260-1630 nm
Chanel Setup time: < 50 ms
Switch reconfiguration time (all ports): 200 ms typical
Polarization Dependent Loss: <0.3 dB
Chromatic dispersion at 1550 nm (EoL): 0.25 ps/nm
Static cross-talk: -60 dB
Dynamic cross-talk: -38 dB
Input Dynamic range: +5 dBm to -20 dBm
Switching cycles: 10^12
Insertion loss (EoL): min 0.8 dB, typical 1.8 dB, max 3.5 dB (O,S,C Bands)
Wavelength dependent loss: 1.0 dB
Return loss (EoL): typical 41 dB, minimum 35 dB

ENVIRONMENTAL
Temperature: Operating 5° to 55° C (41° to 130° F)
Non-operating -40° to +70° C (-40° to 158° F)
Humidity: Operating 10% to 90%, non-condensing
Non-operating 5% to 93%, non-condensing
Altitude: Operating: Up to 1,600 meters
Non-operating: Up to 12,000 meters

POWER
-48v DC dual redundant (A/B) power supplies
Optional Front or Rear mounting of A & B Power Feeds
Field replaceable power modules
Power dissipation: 45 watts typical

MECHANICAL
Size 17.5"w x 12.2"h x 19"d (445 x 310 x 483 mm)
Weight 45 lbs. (20.5kg), Shipping weight 55 lbs. (25kg)

REGULATORY COMPLIANCE
Safety: UL 60950, EN 60950-1, CSA 69950
EMI / EMC: FCC Part 15 Subpart B, GR-1089-CORE, EN 55022, Class A, EN 55024
Environmental: GR-63-CORE (NEBS), EN 300019
Eye safety: CFR Title 21 Part 1040 Class 1
I/P voltage: ANSI T1.315-2001

MANAGEMENT
Interfaces: Dual Gigabit Ethernet Ports, Serial Console Port, External Alarm Contacts
Web GUI
TL1 Command Set, SNMPv3, CORBA, OpenFlow API

AWGR Datasheet

Enablence

NxN AWG MULTIPLEXERS AND DEMULTIPLEXERS ROUTER MODULE
(APRTE)

Enablence's N-by-N arrayed-waveguide grating (AWG) wavelength division multiplexers and demultiplexers are based on our patent-pending CVD process. These silica-on-silicon waveguides exhibit exceptional material uniformity. Complemented with our automated robust packaging, Enablence's planar lightwave circuits (PLC) are well suited for demanding telecom applications such as DWDM, long-haul, and metro transmission systems.

BENEFITS

- 4, 8, 16, 24, and 32CH Capability
- 50 and 200 GHz Channel Spacing
- Custom Packaging
- Choice of Connector and Polish

FEATURES

- Compact and High-Performance WDW Filters
- Low Insertion Loss and Cross Talk
- Can be used as MUX or DMEUX
- High Uniformity and Reliability

APPLICATIONS

- Mesh-Type DWDM Networks
- Wavelength Routing

DATA SHEET

www.enablence.com

Enablence

Enablence's N-by-N arrayed waveguide grating (AWG) can be used in WDM networks with mesh structures. It offers accurate channel alignment, low crosstalk and high channel-to-channel uniformity. In addition, these modules can be used either as muxtiplexer or as demuxtiplexer functions. This product family complies with Telcordia GR-1221-CORE requirements.

CYCLICAL NXN CHANNEL C-BAND AWG ROUTERS OPTICAL SPECIFICATIONS

Parameters		Symbol	Specifications			Units	Comments
			Min	Typ	Max		
Input Channels				N		-	N: 4, 8, 16, 24, 32
Output Channels				N		-	N: 4, 8, 16, 24, 32
Channel Spacing				100		GHz	
Free Spectral Range		FSR		100*N		GHz	Centered at each ITU frequency
Channel Frequencies (Input at (N/2 + 1) to Outputs 1-N)		f_c		C or L-Bands			
ITU Band			-12.50		+12.50	GHz	Centered at each ITU frequency
Wavelength Accuracy	N=32	Δλc	-0.12		+0.12	nm	Offset from ITU grid
	N=24	Δλc	-0.10		+0.10	nm	
	N=16	Δλc	-0.06		+0.06	nm	
	N=8	Δλc	-0.03		+0.03	nm	
	N=4	Δλc	-0.02		+0.02	nm	
Insertion Loss	N=32	IL		7.00		dB	Measured at peak transmission. Measured as 1xN and Nx1 (Input (N/2 +1) to Outputs 1 ~ N And inputs 1 ~ N to output (N/2 +1)
	N=24	IL		6.50		dB	
	N=16	IL		6.00		dB	
	N=8	IL		5.00		dB	
	N=4	IL		5.00		dB	
Insertion Loss Uniformity	N=32	ΔIL		3.00		dB	Any one input to all outputs
	N=24	ΔIL		2.50		dB	
	N=16	ΔIL		2.50		dB	
	N=8	ΔIL		2.00		dB	
	N=4	ΔIL		2.00		dB	
Polarization Dependent Loss		PDL			0.40	dB	Measured at ITU grid frequency
1dB Passband		δ1dB	0.20			nm	Measure 1dB down from min IL
3dB Passband		δ1dB	0.40			nm	Measure 1dB down from min IL
Adjacent Channel Crosstalk		AX			-25.00	dB	At ITU grid frequency
Non-Adjacent Channel Crosstalk		NX			-30.00	dB	At each ITU grid frequency
Total Crosstalk		TX			-22.00	dB	At ITU grid frequency, cumulative sum of all AX and NX
Return Loss		RL	40.00	45.00		dB	

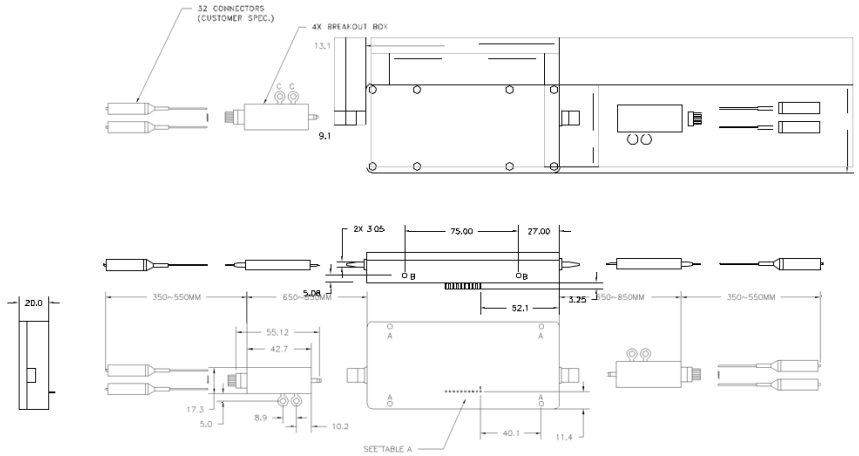


TABLE A: AWG PIN-OUTS OPTIONS

Pin	RTD	Thermistor	ITC
1	Heater +	Heater +	N.C.
2	Heater -	Heater -	+5V
3	RTD1 B1	N.C.	+5V
4	RTD1 B2	Thermistor1	Ready
5	RTD 1 A	Thermistor1	Error / Alarm
6	N.C.	N.C.	Reset / Enable
7	RTD2 A	Thermistor2	TX
8	RTD2 B1	Thermistor2	GND
9	RTD2 B2	N.C.	RX
10	N.C.	N.C.	GND

For more information
visit www.enablence.com

BIBLIOGRAPHY

[1] The green data project. http://www.greendataproject.org/ (visited on 5/10/2016).

[2] Cisco global cloud index: Forecast and methodology, 2014–2019. http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html (visited on 5/10/2016).

[3] Mohd Nazir Shakil. MEMS optical switches. http://www.slideshare.net/hi2mohdnazir/mems-optical-switches (visited on 11/10/2016).

[4] Wavelength routing optical switch. http://www.tsud.elec.keio.ac.jp/english/research/wros_e.html (visited on 11/10/2016).

[5] Odile Liboiron-Ladouceur, Assaf Shacham, Benjamin A. Small, Benjamin G. Lee, Howard Wang, Caroline P. Lai, Aleksandr Biberman, and Keren Bergman. The data vortex optical packet switched interconnection network. *Journal of Lightwave Technology*, 26(13):1777–1789, 2008.

[6] Nathan Farrington, Alex Forencich, Pang-Chen Sun, Shaya Fainman, Joe Ford, Amin Vahdat, George Porter, and George C Papen. A 10 us hybrid optical-circuit/electrical-packet network for datacenters. In *Optical Fiber Communication Conference*, pages OW3H–3. Optical Society of America, 2013.

[7] Kai Chen, Xitao Wen, Xingyu Ma, Yan Chen, Yong Xia, Chengchen Hu, and Qunfeng Dong. Wavecube: A scalable, fault-tolerant, high-performance optical data center architecture. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 1903–1911. IEEE, 2015.

[8] Maria C Yuang, Po-Lung Tien, Hsing-Yu Chen, Wei-Zhang Ruan, Tzu-Kai Hsu, Shan Zhong, Joshua Zhu, Yohann Chen, and Jyehong Chen. Opmdc: Architecture design and implementation of a new optical pyramid data center network. *Journal of Lightwave Technology*, 33(10):2019–2031, 2015.

[9] M.A. Taubenblatt, J.A. Kash, and Y. Taira. Optical interconnects for high performance computing. In *Communications and Photonics Conference and Exhibition (ACP)*, pages 1–2, 2009.

[10] M Farhan Habib, Massimo Tornatore, Marc De Leenheer, Ferhat Dikbiyik, and Biswanath Mukherjee. Design of disaster-resilient optical datacenter networks. *Journal of Lightwave Technology*, 30(16):2563–2573, 2012.

[11] Bikash Koley, Vijay Vusirikala, Cedric Lam, and Vijay Gill. 100GbE and beyond for warehouse scale computing. 2010.

[12] Ken-ichi Sato and Hiroshi Hasegawa. Optical networking technologies that will create future bandwidth-abundant networks [invited]. *Journal of Optical Communications and Networking*, 1(2):A81–A93, 2009.

[13] Masahiko Jinno, Hidehiko Takara, Bartlomiej Kozicki, Yukio Tsukishima, Yoshiaki Sone, and Shinji Matsuoka. Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies. *IEEE Communications Magazine*, 47(11):66–73, 2009.

[14] Charles Despins, Fabrice Labeau, Tho Le Ngoc, Richard Labelle, Mohamed Cheriet, Claude Thibeault, Francois Gagnon, Alberto Leon-Garcia, Omar Cherkaoui, Bill St Arnaud, et al. Leveraging green communications for carbon emission reductions: Techniques, testbeds, and emerging carbon footprint standards. *IEEE Communications Magazine*, 49(8):101–109, 2011.

[15] Alan Benner. Optical interconnect opportunities in supercomputers and high end computing. In *Optical Fiber Communication Conference*, pages OTu2B–4. Optical Society of America, 2012.

[16] Cisco Nexus 3064-X, 3064-T, and 3064-32T Switches. http://www.cisco.com/c/en/us/products/collateral/switches/nexus-3000-series-switches/data_sheet_c78-651097.pdf (visited on 21/4/2017).

[17] Cisco Catalyst 4500-X Series. http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-4500-x-series-switches/data_sheet_c78-696791.html (visited on 21/4/2017).

[18] SMART 2020: enabling the low carbon economy in the information age. http://gesi.org/files/Reports/Smart%202020%20report%20in%20English.pdf (visited on 25/03/2017), 2008.

[19] Ashok V. Krishnamoorthy. The intimate integration of photonics and electronics. In *Advances in Information Optics and Photonics*, volume 1, page 581, 2008.

[20] Here comes the terabit per second network. http://www.zdnet.com/article/here-comes-the-terabit-per-second-network (visited on 5/10/2016).

[21] Inside ten of the worldŠs largest data centers. http://wikibon.org/blog/inside-ten-of-the-worlds-largest-data-centers/ (visited on 12/10/2016).

[22] James Hamilton et al. Data center networks are in my way. *Standford Clean Slate CTO Summit*, 2009.

[23] Nathan Farrington, George Porter, Sivasankar Radhakrishnan, Hamid Hajabdolali Bazzaz, Vikram Subramanya, Yeshaiahu Fainman, George Papen, and Amin Vahdat. Helios: a hybrid electrical/optical switch architecture

for modular data centers. *ACM SIGCOMM Computer Communication Review*, 41(4):339–350, 2011.

[24] Alcatel-lucent breaks new world record for undersea data transmission. http://www3.alcatel-lucent.com/wps/portal/!ut/p/kcxml/04_Sj9SPykssy0xPLMnMz0vM0Y_QjzKLd4x3tXDUL8h2VAQAURh_Yw!!?LMSG_CABINET=Docs_and_Resource_Ctr&LMSG_CONTENT_FILE=News_Releases_2013/News_Article_002876.xml (visited on 5/10/2016).

[25] Cedric Lam, Hong Liu, Bikash Koley, Xiaoxue Zhao, Valey Kamalov, and Vijay Gill. Fiber optic communication technologies: What's needed for datacenter network operations. *Communications Magazine, IEEE*, 48(7):32–39, 2010.

[26] Photonic optical circuit switching | CALIENT technologies. http://www.calient.net/ (visited on 5/10/2016).

[27] Yawei Yin, Roberto Proietti, Xiaohui Ye, Christopher J Nitta, Venkatesh Akella, and SJB Yoo. LIONS: an AWGR-based low-latency optical switch for high-performance computing and data centers. *Selected Topics in Quantum Electronics, IEEE Journal of*, 19(2):3600409–3600409, 2013.

[28] Slavisa Aleksic. Analysis of power consumption in future high-capacity network nodes. *Optical Communications and Networking, IEEE/OSA Journal of*, 1(3):245–258, 2009.

[29] Odile Liboiron-Ladouceur, Isabella Cerutti, Pier Giorgio Raponi, Nicola Andriolli, and Piero Castoldi. Energy-efficient design of a scalable optical multiplane interconnection architecture. *Selected Topics in Quantum Electronics, IEEE Journal of*, 17(2):377–383, 2011.

[30] FINISAR WSS datasheet & application note - datasheet archive. http://www.datasheetarchive.com/FINISAR%20WSS-datasheet.html (visited on 5/10/2016).

[31] George Porter, Richard Strong, Nathan Farrington, Alex Forencich, Pang Chen-Sun, Tajana Rosing, Yeshaiahu Fainman, George Papen, and Amin Vah-

dat. *Integrating microsecond circuit switching into the data center*, volume 43. ACM, 2013.

[32] Imrich Chlamtac, Aura Ganz, and Gadi Karmi. Lightpath communications: An approach to high bandwidth optical WAN's. *Communications, IEEE Transactions on*, 40(7):1171–1182, 1992.

[33] Rajiv Ramaswami and Kumar N Sivarajan. Routing and wavelength assignment in all-optical networks. *IEEE/ACM Transactions on Networking (TON)*, 3(5):489–500, 1995.

[34] Mike J O'Mahony, Dimitra Simeonidou, David K Hunter, and Anna Tzanakaki. The application of optical packet switching in future communication networks. *Communications Magazine, IEEE*, 39(3):128–135, 2001.

[35] George N Rouskas and Lisong Xu. Optical packet switching. In *Emerging Optical Network Technologies*, pages 111–127. Springer, 2005.

[36] Daniel J BLUMENTHAL. Optical packet switching. In *Lasers and Electro-optics Society*, 2004.

[37] Richard Epworth. Optical packet switching, September 30 2003. US Patent 6,626,589.

[38] Chunming Qiao and Myungsik Yoo. Optical burst switching (OBS)–a new paradigm for an optical internet. *Journal of high speed networks*, 8(1):69–84, 1999.

[39] Yang Chen, Chunming Qiao, and Xiang Yu. Optical burst switching: a new area in optical networking research. *Network, IEEE*, 18(3):16–23, 2004.

[40] Tzvetelina Battestilli and Harry Perros. An introduction to optical burst switching. *IEEE communications magazine*, 41(8):S10–S15, 2003.

[41] An Ge, Franco Callegati, and Lakshman S Tamil. On optical burst switching and self-similar traffic. *IEEE Communications Letters*, 4(3):98–100, 2000.

[42] Yijun Xiong, Marc Vandenhoute, and Hakki C. Cankaya. Control architecture in optical burst-switched WDM networks. *Selected Areas in Communications, IEEE Journal on*, 18(10):1838–1851, 2000.

[43] Tao Zhang. A framework for fiber delay-line buffers in packet-based asynchronous multifiber optical networks (PAMFONET). *International Journal of Communication Systems*, 25(2):158–168, 2012.

[44] Xiaomin Lu and Brian L Mark. Performance modeling of optical-burst switching with fiber delay lines. *IEEE Transactions on Communications*, 52(12):2175–2183, 2004.

[45] Yang Chen, Hongyi Wu, Dahai Xu, and Chunming Qiao. Performance analysis of optical burst switched node with deflection routing. In *Communications, 2003. ICC'03. IEEE International Conference on*, volume 2, pages 1355–1359, 2003.

[46] Sungchang Kim, Namook Kim, and Minho Kang. Contention resolution for optical burst switching networks using alternative routing. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 5, pages 2678–2681, 2002.

[47] Jeyashankher Ramamirtham, Jonathan Turner, and Joel Friedman. Design of wavelength converting switches for optical burst switching. *Selected Areas in Communications, IEEE Journal on*, 21(7):1122–1132, 2003.

[48] Vinod M. Vokkarane, Jason P. Jue, and Sriranjani Sitaraman. Burst segmentation: an approach for reducing packet loss in optical burst switched networks. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 5, pages 2673–2677, 2002.

[49] Vinod M Vokkarane and Jason P Jue. Prioritized burst segmentation and composite burst-assembly techniques for qos support in optical burst-switched networks. *IEEE journal on Selected Areas in Communications*, 21(7):1198–1209, 2003.

[50] Guohui Wang, David G. Andersen, Michael Kaminsky, Konstantina Papagiannaki, T. S. Ng, Michael Kozuch, and Michael Ryan. c-Through: Part-time optics in data centers. In *ACM SIGCOMM Computer Communication Review*, volume 40, pages 327–338, 2010.

[51] Wang Miao, Fernando Agraz, Shuping Peng, Salvatore Spadaro, Giacomo Bernini, Jordi Perelló, Georgios Zervas, Reza Nejabati, Nicola Ciulli, Dimitra Simeonidou, et al. SDN-enabled OPS with QoS guarantee for reconfigurable virtual data center networks. *Journal of Optical Communications and Networking*, 7(7):634–643, 2015.

[52] Shuping Peng, Bingli Guo, Chris Jackson, Reza Nejabati, Fernando Agraz, Salvatore Spadaro, Giacomo Bernini, Nicola Ciulli, and Dimitra Simeonidou. Multi-tenant software-defined hybrid optical switched data centre. *Journal of Lightwave Technology*, 33(15):3224–3233, 2015.

[53] K. Christodoulopoulos, D. Lugones, K. Katrinis, M. Ruffini, and D. O'Mahony. Performance evaluation of a hybrid optical/electrical interconnect. *Optical Communications and Networking, IEEE/OSA Journal of*, 7(3):193–204, March 2015.

[54] Kai Chen, A. Singla, A. Singh, K. Ramachandran, Lei Xu, Yueping Zhang, Xitao Wen, and Yan Chen. OSA: An optical switching architecture for data center networks with unprecedented flexibility. *Networking, IEEE/ACM Transactions on*, 22(2):498–511, April 2014.

[55] Lih Y. Lin, Evan L. Goldstein, and Robert W. Tkach. Free-space micromachined optical switches for optical networking. *Selected Topics in Quantum Electronics, IEEE Journal of*, 5(1):4–9, 1999.

[56] Series 7000 - 384x384 port software-defined optical circuit switch. `http://www.polatis.com/series-7000-384x384-port-software-controlled-optical-circuit-switch-asp` (visited on 5/4/2017).

[57] J. Kim, C. J. Nuzman, B. Kumar, D. F. Lieuwen, J. S. Kraus, A. Weiss, C. P. Lichtenwalner, A. R. Papazian, R. E. Frahm, and N. R. Basavanhally. 1100 x 1100 port MEMS-based optical crossconnect with 4-dB maximum loss. *Photonics Technology Letters, IEEE*, 15(11):1537–1539, 2003.

[58] Y. Hida, Y. Hibino, T. Kitoh, Y. Inoue, M. Itoh, T. Shibata, A. Sugita, and A. Himeno. 400-channel arrayed-waveguide grating with 25 GHz spacing using 1.5%-δ waveguides on 6-inch si wafer. *Electronics Letters*, 37(9):576–577, 2001.

[59] K. Takada, M. Abe, M. Shibata, M. Ishii, and K. Okamoto. Low-crosstalk 10-GHz-spaced 512-channel arrayed-waveguide grating multi/demultiplexer fabricated on a 4-in wafer. *Photonics Technology Letters, IEEE*, 13(11):1182–1184, 2001.

[60] Stanley Cheung, Tiehui Su, Katsunari Okamoto, and SJB Yoo. Ultra-compact silicon photonic 512× 512 25 ghz arrayed waveguide grating router. *IEEE Journal of Selected Topics in Quantum Electronics*, 20(4):310–316, 2014.

[61] NxN AWG multiplexers and demultiplexers router module. `http://www.enablence.com/media/pdfs/Datasheet_OCSD_AWG_Other_NxN_APRTE_0.pdf` (visited on 5/4/2017).

[62] NxN AWG multiplexers and demultiplexers router module. `http://www.wooriro.com/img/product/spec-plc/(wos-f-1901)_nxn_awg.pdf` (visited on 5/4/2017).

[63] Awg multi/demultiplexer. `http://www.ntt-electronics.com/en/products/photonics/awg_mul_d.html` (visited on 5/4/2017).

[64] K Nashimoto, D Kudzuma, and H Han. High-speed switching and filtering using PLZT waveguide devices. In *OptoeElectronics and Communications Conference (OECC), 2010 15th*, pages 540–542. IEEE, 2010.

[65] Ibrahim Murat Soganci, Takuo Tanemura, KA Williams, Nicola Calabretta, T de Vries, E Smalbrugge, MK Smit, HJS Dorren, and Yoko Nakano. High-

speed 1x16 optical switch monolithically integrated on InP. *ECOC 2009*, 2009.

[66] Epiphotonics. `http://epiphotonics.com/` (visited on 5/10/2016).

[67] R. Stabile, A. Albores-Mejia, and K. A. Williams. Monolithic active-passive 16 × 16 optoelectronic switch. *Opt. Lett.*, 37(22):4666–4668, Nov 2012.

[68] H. Wang, A. Wonfor, K. A. Williams, R. V. Penty, and I. H. White. Demonstration of a lossless monolithic 16x16 QW SOA switch. In *2009 35th European Conference on Optical Communication*, volume 2009-Supplement, pages 1–2, Sept 2009.

[69] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of mathematics*, 17(3):449–467, 1965.

[70] Kai Chen, Ankit Singlay, Atul Singhz, Kishore Ramachandranz, Lei Xuz, Yueping Zhangz, Xitao Wen, and Yan Chen. OSA: an optical switching architecture for data center networks with unprecedented flexibility. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 18–18, Berkeley, CA, USA, 2012. USENIX Association.

[71] Diego Lugones, Kostas Katrinis, and Martin Collier. A reconfigurable optical/-electrical interconnect architecture for large-scale clusters and datacenters. In *Proceedings of the 9th conference on Computing Frontiers*, pages 13–22, 2012.

[72] Diego Lugones, Kostas Katrinis, Georgios Theodoropoulos, and Martin Collier. A reconfigurable, regular-topology cluster/datacenter network using commodity optical switches. *Future Generation Computer Systems*, 30:78–89, 2014.

[73] Roe Hemenway, Richard Grzybowski, Cyriel Minkenberg, and Ronald Luijten. Optical-packet-switched interconnect for supercomputer applications. *Journal of Optical Networking*, 3(12):900–913, 2004.

[74] Ronald Luijten, Wolfgang E Denzel, Richard R Grzybowski, and Roe Hemenway. Optical interconnection networks: The OSMOSIS project. In *The 17th Annual Meeting of the IEEE Lasers and Electro-Optics Society*, 2004.

[75] Ronald Luijten and Richard Grzybowski. The OSMOSIS optical packet switch for supercomputers. In *Optical Fiber Communication Conference*, page OTuF3. Optical Society of America, 2009.

[76] Cyriel Minkenberg, Ilias Iliadis, and François Abel. Low-latency pipelined crossbar arbitration. In *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, volume 2, pages 1174–1179, 2004.

[77] Cory Hawkins, Benjamin A. Small, D. Scott Wills, and Keren Bergman. The data vortex, an all optical path multicomputer interconnection network. *Parallel and Distributed Systems, IEEE Transactions on*, 18(3):409–420, 2007.

[78] Keren Bergman and Howard Wang. Optically interconnected high performance data centers. In *Optical Interconnects for Future Data Center Networks*, pages 155–167. Springer, 2013.

[79] Assaf Shacham and Keren Bergman. An experimental validation of a wavelength-striped, packet switched, optical interconnection network. *Journal of Lightwave Technology*, 27(7):841–850, 2009.

[80] Howard Wang and Keren Bergman. A bidirectional 2x2 photonic network building-block for high-performance data centers. In *Optical Fiber Communication Conference and Exposition (OFC/NFOEC), 2011 and the National Fiber Optic Engineers Conference*, pages 1–3, 2011.

[81] Odile Liboiron-Ladouceur, Pier Giorgio Raponi, Nicola Andriolli, Isabella Cerutti, Mohammed Shafiqul Hai, and Piero Castoldi. A scalable space–time multi-plane optical interconnection network using energy-efficient enabling technologies [invited]. *Optical Communications and Networking, IEEE/OSA Journal of*, 3(8):A1–A11, 2011.

[82] Isabella Cerutti, Pier Giorgio Raponi, Nicola Andriolli, Piero Castoldi, and Odile Liboiron-Ladouceur. Designing energy-efficient data center networks using space-time optical interconnection architectures. *Selected Topics in Quantum Electronics, IEEE Journal of*, 19(2):3700209–3700209, 2013.

[83] Xiaohui Ye, Yawei Yin, S. J. B. Yoo, Paul Mejia, Roberto Proietti, and Venkatesh Akella. DOS: a scalable optical switch for datacenters. In *Proceedings of the 6th ACM/IEEE Symposium on Architectures for Networking and Communications Systems*, ANCS '10, pages 24:1–24:12, New York, NY, USA, 2010. ACM.

[84] SJ Ben Yoo. Optical packet and burst switching technologies for the future photonic internet. *Lightwave Technology, Journal of*, 24(12):4468–4492, 2006.

[85] Roberto Proietti, Xiaohui Ye, Yawei Yin, Andrew Potter, Runxiang Yu, Junya Kurumida, Venkatesh Akella, and S. J. Ben Yoo. 40 Gb/s 8x8 Low-latency optical switch for data centers. In *Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2011*, OSA Technical Digest (CD), page OMV4. Optical Society of America, March 2011.

[86] Roberto Proietti, Yawei Yin, Runxiang Yu, Xiaohui Ye, Christopher Nitta, Venkatesh Akella, and SJ Ben Yoo. All-optical physical layer NACK in AWGR-based optical interconnects. *Photonics Technology Letters, IEEE*, 24(5):410–412, 2012.

[87] Roberto Proietti, Christopher J Nitta, Yawei Yin, Runxiang Yu, SJB Yoo, and Venkatesh Akella. Scalable and distributed contention resolution in AWGR-based data center switches using RSOA-based optical mutual exclusion. *Selected Topics in Quantum Electronics, IEEE Journal of*, 19(2):3600111–3600111, 2013.

[88] Zheng Cao, Roberto Proietti, and SJB Yoo. Hi-LION: Hierarchical large-scale interconnection optical network with AWGRs [invited]. *Journal of Optical Communications and Networking*, 7(1):A97–A105, 2015.

[89] Roberto Proietti, Yawei Yin, Runxiang Yu, Christopher J Nitta, Venkatesh Akella, Christopher Mineo, and SJ Yoo. Scalable optical interconnect architecture using AWGR-Based TONAK LION switch with limited number of wavelengths. *Journal of Lightwave Technology*, 31(24):4087–4097, 2013.

[90] Kang Xia, Yu-Hsiang Kaob, Ming Yangb, and HJ Chao. Petabit optical switch for data center networks. *Polytechnic Institute of New York University, New York, Tech. Rep*, 2010.

[91] Jurgen Gripp, J. E. Simsarian, J. D. LeGrange, P. Bernasconi, and D. T. Neilson. Photonic terabit routers: the IRIS project. In *Optical Fiber Communication Conference*, 2010.

[92] J Gripp, D Stiliadis, JE Simsarian, P Bernasconi, JD Le Grange, L Zhang, L Buhl, and DT Neilson. Iris optical packet router [invited]. *Journal of Optical Networking*, 5(8):589–597, 2006.

[93] Philip N Ji, Dayou Qian, Konstantinos Kanonakis, Christoforos Kachris, and Ioannis Tomkos. Design and evaluation of a flexible-bandwidth OFDM-based intra-data center interconnect. *Selected Topics in Quantum Electronics, IEEE Journal of*, 19(2):3700310–3700310, 2013.

[94] Yuanqiu Luo, Jianjun Yu, Junqiang Hu, Lei Xu, Philip N Ji, Ting Wang, and Milorad Cvijetic. Wdm passive optical network with parallel signal detection for video and data delivery. In *Optical Fiber Communication Conference*, page OWS6. Optical Society of America, 2007.

[95] Philip N Ji, Ting Wang, Dayou Qian, Lei Xu, Yoshiaki Aono, Tsutomu Tajima, Christoforos Kachris, Konstantinos Kanonakis, Ioannis Tomkos, Tiejun J Xia, et al. Demonstration of high-speed mimo ofdm flexible bandwidth data center network. In *European Conference and Exhibition on Optical Communication*, pages Th–2. Optical Society of America, 2012.

[96] Jordi Perelló, Salvatore Spadaro, Sergio Ricciardi, Davide Careglio, Shuping Peng, Reza Nejabati, Georgios Zervas, Dimitra Simeonidou, Alessandro

Predieri, Matteo Biancani, et al. All-optical packet/circuit switching-based data center network for enhanced scalability, latency, and throughput. *Network, IEEE*, 27(6):14–22, 2013.

[97] Shuping Peng, Dimitra Simeonidou, Georgios Zervas, Reza Nejabati, Yan Yan, Yi Shu, Salvatore Spadaro, Jordi Perello, Fernando Agraz, Davide Careglio, et al. A novel SDN enabled hybrid optical packet/circuit switched data centre network: The LIGHTNESS approach. In *Networks and Communications (EuCNC), 2014 European Conference on*, pages 1–5. IEEE, 2014.

[98] JM Saridis, Shpuping Peng, Yan Yan, Alejandro Aguado, Bingli Buo, Murat Arslan, Chris Jackson, Wang Miao, Nicola Calabretta, Fernando Agraz Bujan, et al. LIGHTNESS: a deeply-programmable SDN-enabled data centre network with OCS/OPS multicast/unicast switch-over. In *2015 European Conference on Optical Communication (ECOC 2015): Valencia, Spain: 27 September-1 October 2015*, pages 1–3. Institute of Electrical and Electronics Engineers (IEEE), 2015.

[99] George M Saridis, Shuping Peng, Yan Yan, Alejandro Aguado, Bingli Guo, Murat Arslan, Chris Jackson, Wang Miao, Nicola Calabretta, Fernando Agraz, et al. LIGHTNESS: a function-virtualizable software defined data center network with all-optical circuit/packet switching. *Journal of Lightwave Technology*, 34(7):1618–1627, 2016.

[100] Matteo Fiorani, Slavisa Aleksic, and Maurizio Casoni. Hybrid optical switching for data center networks. *Journal of Electrical and Computer Engineering*, 2014, 2014.

[101] Matteo Fiorani, Maurizio Casoni, and Slavisa Aleksic. Performance and power consumption analysis of a hybrid optical core node. *Optical Communications and Networking, IEEE/OSA Journal of*, 3(6):502–513, 2011.

[102] Kevin J Barker, Alan Benner, Ray Hoare, Adolfy Hoisie, Alex K Jones, Darren K Kerbyson, Dan Li, Rami Melhem, Ram Rajamony, Eugen Schenfeld,

et al. On the feasibility of optical circuit switching for high performance computing systems. In *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, page 16. IEEE Computer Society, 2005.

[103] Cisco nexus 5596. http://www.cisco.com/c/en/us/products/collateral/switches/nexus-5548p-switch/white_paper_c11-622479.html (visited on 5/10/2016).

[104] Cisco nexus 5548P, 5548UP, 5596UP, and 5596T switches data sheet. http://www.cisco.com/c/en/us/products/collateral/switches/nexus-5000-series-switches/data_sheet_c78-618603.html (visited on 5/10/2016).

[105] Cisco GPL 2017. http://itprice.com/cisco-gpl/3K-C3064-X-FA-L3 (visited on 21/4/2017).

[106] Cisco GPL 2017. http://itprice.com/cisco-gpl/WS-C4500X-32SFP (visited on 21/4/2017).

[107] Photonic optical circuit switching | CALIENT technologies. https://www.phoenixdatacom.com/wp-content/uploads/2015/11/Calient-S320-Datasheet-Nov6-2014-pdl.pdf (visited on 5/10/2016).

[108] Cisco SFP-10G-ZR. http://www.fs.com/products/11558.html (visited on 5/10/2016).

[109] Cisco S-Class 10GBASE SFP+ Modules Data Sheet. http://www.cisco.com/c/en/us/products/collateral/interfaces-modules/transceiver-modules/datasheet-c78-733585.html (visited on 5/10/2016).

[110] Tunable Laser Cost. https://www.arrow.com/en/products/ftlx6624mcc/finisar (visited on 21/4/2017).

[111] Tunable Laser. https://www.finisar.com/sites/default/files/downloads/finisar_ftlx6871mcc_ftlx6872mcc_10g_dwdm_80km_

multi_rate_tunable_sfp_transceiver_productspecb1_1.pdf (visited on 21/4/2017).

[112] FPGA Cost. https://www.arrow.com/en/products/10ax027e3f29e2sg/alteraintel-programmable-solutions (visited on 21/4/2017).

[113] FPGA DataSheet. http://static6.arrow.com/aropdfconversion/b99a4c1ec2bfbaef5278785a345d10af1d5eff91/41999112980983904a10_over.pdf (visited on 21/4/2017).

[114] Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. A scalable, commodity data center network architecture. *ACM SIGCOMM Computer Communication Review*, 38(4):63–74, 2008.

[115] Average retail electricity prices in the u.s. from 1990 to 2015 (in cents per kilowatt hour). https://www.statista.com/statistics/183700/us-average-retail-electricity-price-since-1990/ (visited on 16/10/2016).

[116] Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi, Chen Tian, Yongguang Zhang, and Songwu Lu. BCube: a high performance, server-centric network architecture for modular data centers. *ACM SIGCOMM Computer Communication Review*, 39(4):63–74, 2009.

[117] OMNeT++ simulation framework. http://omnetpp.org/ (visited on 5/10/2016).

[118] INET++ framework. https://inet.omnetpp.org/Introduction.html (visited on 16/10/2016).

[119] Srikanth Kandula, Sudipta Sengupta, Albert Greenberg, Parveen Patel, and Ronnie Chaiken. The nature of data center traffic: measurements & analysis. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 202–208, 2009.

[120] Srikanth Kandula, Jitendra Padhye, and Paramvir Bahl. Flyways to de-congest data center networks. 2009.

[121] Theophilus Benson, Aditya Akella, and David A. Maltz. Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 267–280, 2010.

[122] Theophilus Benson, Ashok Anand, Aditya Akella, and Ming Zhang. Understanding data center traffic characteristics. *ACM SIGCOMM Computer Communication Review*, 40(1):92–99, 2010.

[123] Balakrishnan Chandrasekaran. Survey of network traffic models. *Waschington University in St. Louis CSE*, 567, 2009.

[124] Markov chain. `https://en.wikipedia.org/wiki/Markov_chain` (visited on 5/10/2016).

[125] Markov chain. `https://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/Chapter11.pdf` (visited on 5/10/2016).

[126] Tuan Phung-Duc, Hiroyuki Masuyama, Shoji Kasahara, and Yutaka Takahashi. Performance analysis of optical burst switched networks with limited-range wavelength conversion, retransmission and burst segmentation. *Journal of the Operations Research Society of Japan*, 52(1):58–74, 2009.

[127] Andrea Detti, Vincenzo Eramo, and M Listanti. Performance evaluation of a new technique for ip support in a WDM optical network: optical composite burst switching (OCBS). *Journal of Lightwave Technology*, 20(2):154, 2002.

[128] Nail Akar, Ezhan Karasan, and Kaan Dogan. Wavelength converter sharing in asynchronous optical packet/burst switching: an exact blocking analysis for markovian arrivals. *IEEE Journal on Selected Areas in Communications*, 24(12):69–80, 2006.

[129] Ayman Malek Kaheel, Hussein Alnuweiri, and Fayez Gebali. A new analytical model for computing blocking probability in optical burst switching networks. *IEEE Journal on Selected Areas in Communications*, 24(12):120–128, 2006.

[130] David Erickson. The beacon openflow controller. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 13–18. ACM, 2013.

[131] Christoforos Kachris and Ioannis Tomkos. A survey on optical interconnects for data centers. *Communications Surveys & Tutorials, IEEE*, 14(4):1021–1036, 2012.

[132] Muhammad Imran, Pascal Landais, Martin Collier, and Kostas Katrinis. Performance analysis of optical burst switching with fast optical switches for data center networks. In *2015 17th International Conference on Transparent Optical Networks (ICTON)*, pages 1–4. IEEE, 2015.

[133] Jin Heo and Lenin Singaravelu. Deploying extremely latency-sensitive applications in vSphere 5.5: Performance study. Technical report, Technical report, VMware, Inc., 2013. www. vmware. com/files/pdf/techpaper/latency-sensitive-perf-vsphere55. pdf.

[134] Cut-through and store-and-forward ethernet switching for low-latency environments. `http://www.cisco.com/c/en/us/products/collateral/switches/nexus-5020-switch/white_paper_c11-465436.html` (visited on 5/10/2016).

[135] Transmission Control Protocol. `https://www.rfc-editor.org/rfc/rfc793.txt` (visited on 5/10/2016).

[136] TCP Extensions for High Performance. `https://www.ietf.org/rfc/rfc1323.txt` (visited on 5/10/2016).

[137] TCP Extensions for High Performance. `https://www.ietf.org/rfc/rfc1323.txt` (visited on 5/10/2016).

[138] Sunil Gowda, Ramakrishna K Shenai, Krishna M Sivalingam, and Hakki Candan Cankaya. Performance evaluation of TCP over optical burst-switched (OBS) WDM networks. In *Communications, 2003. ICC'03. IEEE International Conference on*, volume 2, pages 1433–1437. IEEE, 2003.

[139] Xiang Yu, Jikai Li, Xiaojun Cao, Yang Chen, and Chunming Qiao. Traffic statistics and performance evaluation in optical burst switched networks. *Lightwave Technology, Journal of*, 22(12):2722–2738, 2004.

[140] Xiang Yu, Chunming Qiao, and Yong Liu. TCP implementations and false time out detection in OBS networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 774–784. IEEE, 2004.

[141] Qiong Zhang, Vinod M Vokkarane, Yuke Wang, and Jason P Jue. Analysis of TCP over optical burst-switched networks with burst retransmission. In *Global Telecommunications Conference, 2005. GLOBECOM'05. IEEE*, volume 4, pages 6–pp. IEEE, 2005.

[142] L Zhu, N Ansari, and J Liu. Throughput of high-speed TCP in optical burst switching networks. *IEE Proceedings-Communications*, 152(3):349–352, 2005.

[143] Basem Shihada, Qiong Zhang, Pin-Han Ho, and Jason P Jue. A novel implementation of TCP Vegas for optical burst switched networks. *Optical switching and networking*, 7(3):115–126, 2010.

[144] Bharat Komatireddy, Neal Charbonneau, and Vinod M Vokkarane. Source-ordering for improved TCP performance over load-balanced optical burst-switched (OBS) networks. *Photonic Network Communications*, 19(1):1–8, 2010.

[145] Subhasis Datta, Avijan Dutta, and Subhrabrata Choudhury. Design and analysis of a modified TCP for optical burst switched networks. In *Business and Information Management (ICBIM), 2014 2nd International Conference on*, pages 7–10. IEEE, 2014.

[146] Lei Liu, Hongxiang Guo, Takehiro Tsuritani, Yawei Yin, Jian Wu, Xiaobin Hong, Jintong Lin, and Masatoshi Suzuki. Dynamic provisioning of self-organized consumer grid services over integrated OBS/WSON networks. *Journal of Lightwave Technology*, 30(5):734–753, 2012.

[147] Kostas Ramantas and Kyriakos Vlachos. A TCP-specific traffic profiling and prediction scheme for performance optimization in OBS networks. *Journal of Optical Communications and Networking*, 3(12):924–936, 2011.

[148] Shuping Peng, Zhengbin Li, Yongqi He, and Anshi Xu. TCP window-based flow-oriented dynamic assembly algorithm for OBS networks. *Journal of Lightwave Technology*, 27(6):670–678, 2009.

[149] Sally Floyd. Highspeed TCP for large congestion windows. 2003.

[150] N Sreenath, N Srinath, J Aloysius Suren, and KDSSU Kumar. Reducing the impact of false time out on TCP performance in TCP over OBS networks. *Photonic Network Communications*, 27(1):47–56, 2014.

[151] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008.